® 2024 ANSYS, Inc. or its affiliated companies
Unauthorized use, distribution, or duplication is prohibited.

Ansys Fluent Theory Guide

Ansys

ANSYS, Inc.
Southpointe
2600 Ansys Drive
Canonsburg, PA 15317
ansysinfo@ansys.com
http://www.ansys.com
(T) 724-746-3304
(F) 724-514-9494

Release 2024 R1
January 2024

ANSYS, Inc. and ANSYS Europe, Ltd. are UL registered ISO 9001: 2015 companies.

Copyright and Trademark Information

© 2024 ANSYS, Inc. Unauthorized use, distribution or duplication is prohibited.

ANSYS, Ansys Workbench, AUTODYN, CFX, FLUENT and any and all ANSYS, Inc. brand, product, service and feature names, logos and slogans are registered trademarks or trademarks of ANSYS, Inc. or its subsidiaries located in the United States or other countries. ICEM CFD is a trademark used by ANSYS, Inc. under license. CFX is a trademark of Sony Corporation in Japan. All other brand, product, service and feature names or trademarks are the property of their respective owners. FLEXlm and FLEXnet are trademarks of Flexera Software LLC.

Disclaimer Notice

THIS ANSYS SOFTWARE PRODUCT AND PROGRAM DOCUMENTATION INCLUDE TRADE SECRETS AND ARE CONFIDENTIAL AND PROPRIETARY PRODUCTS OF ANSYS, INC., ITS SUBSIDIARIES, OR LICENSORS. The software products and documentation are furnished by ANSYS, Inc., its subsidiaries, or affiliates under a software license agreement that contains provisions concerning non-disclosure, copying, length and nature of use, compliance with exporting laws, warranties, disclaimers, limitations of liability, and remedies, and other provisions. The software products and documentation may be used, disclosed, transferred, or copied only in accordance with the terms and conditions of that software license agreement.

ANSYS, Inc. and ANSYS Europe, Ltd. are UL registered ISO 9001: 2015 companies.

U.S. Government Rights

For U.S. Government users, except as specifically granted by the ANSYS, Inc. software license agreement, the use, duplication, or disclosure by the United States Government is subject to restrictions stated in the ANSYS, Inc. software license agreement and FAR 12.212 (for non-DOD licenses).

Third-Party Software

See the legal information in the product help files for the complete Legal Notice for ANSYS proprietary software and third-party software. If you are unable to access the Legal Notice, contact ANSYS, Inc.

Published in the U.S.A.

Table of Contents

Using This Manual xli
1. The Contents of This Manual xli
2. Typographical Conventions xlii
3. Mathematical Conventions xlv
1. Basic Fluid Flow 1
1.1. Overview of Physical Models in Ansys Fluent 1
1.2. Continuity and Momentum Equations 2
1.2.1. The Mass Conservation Equation 3
1.2.2. Momentum Conservation Equations 3
1.3. User-Defined Scalar (UDS) Transport Equations 4
1.3.1. Single Phase Flow 4
1.3.2. Multiphase Flow 5
1.4. Periodic Flows 6
1.4.1. Overview 6
1.4.2. Limitations 7
1.4.3. Physics of Periodic Flows 7
1.4.3.1. Definition of the Periodic Velocity 7
1.4.3.2. Definition of the Streamwise-Periodic Pressure 8
1.5. Swirling and Rotating Flows 9
1.5.1. Overview of Swirling and Rotating Flows 9
1.5.1.1. Axisymmetric Flows with Swirl or Rotation 9
1.5.1.1.1. Momentum Conservation Equation for Swirl Velocity 10
1.5.1.2. Three-Dimensional Swirling Flows 11
1.5.1.3. Flows Requiring a Moving Reference Frame 11
1.5.2. Physics of Swirling and Rotating Flows 11
1.6. Compressible Flows 12
1.6.1. When to Use the Compressible Flow Model 14
1.6.2. Physics of Compressible Flows 14
1.6.2.1. Basic Equations for Compressible Flows 15
1.6.2.2. The Compressible Form of the Gas Law 15
1.7. Inviscid Flows 15
1.7.1. Euler Equations 16
1.7.1.1. The Mass Conservation Equation 16
1.7.1.2. Momentum Conservation Equations 16
1.7.1.3. Energy Conservation Equation 17
2. Flows with Moving Reference Frames 19
2.1. Introduction 19
2.2. Flow in a Moving Reference Frame 21
2.2.1. Equations for a Moving Reference Frame 21
2.2.1.1. Relative Velocity Formulation 22
2.2.1.2. Absolute Velocity Formulation 23
2.2.1.3. Relative Specification of the Reference Frame Motion 23
2.3. Flow in Multiple Reference Frames 24
2.3.1. The Multiple Reference Frame Model 24
2.3.1.1. Overview 24
2.3.1.2. Examples 25
2.3.1.3. The MRF Interface Formulation 26
2.3.1.3.1. Interface Treatment: Relative Velocity Formulation 26
2.3.1.3.2. Interface Treatment: Absolute Velocity Formulation 27

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

2.3.2. The Mixing Plane Model 28
2.3.2.1. Overview 28
2.3.2.2. Rotor and Stator Domains 28
2.3.2.3. The Mixing Plane Concept 30
2.3.2.4. Choosing an Averaging Method 30
2.3.2.4.1. Area Averaging 31
2.3.2.4.2. Mass Averaging 31
2.3.2.4.3. Mixed-Out Averaging 31
2.3.2.5. Mixing Plane Algorithm of Ansys Fluent 32
2.3.2.6. Mass Conservation 32
2.3.2.7. Swirl Conservation 33
2.3.2.8. Total Enthalpy Conservation 34
3. Flows Using Sliding and Dynamic Meshes 35
3.1. Introduction 35
3.2. Dynamic Mesh Theory 37
3.2.1. Conservation Equations 38
3.2.2. Six DOF Solver Theory 39
3.3. Sliding Mesh Theory 40
4. Turbulence 41
4.1. Underlying Principles of Turbulence Modeling 41
4.1.1. Reynolds (Ensemble) Averaging 42
4.1.2. Filtered Navier-Stokes Equations 42
4.1.3. Hybrid RANS-LES Formulations 43
4.1.4. Boussinesq Approach vs. Reynolds Stress Transport Models 44
4.2. Spalart-Allmaras Model 44
4.2.1. Overview 45
4.2.2. Transport Equation for the Spalart-Allmaras Model 45
4.2.3. Modeling the Turbulent Viscosity 46
4.2.4. Modeling the Turbulent Production 46
4.2.5. Modeling the Turbulent Destruction 47
4.2.6. Model Constants 47
4.2.7. Wall Boundary Conditions 47
4.2.7.1. Treatment of the Spalart-Allmaras Model for Icing Simulations 48
4.2.8. Convective Heat and Mass Transfer Modeling 48
4.3. Standard, RNG, and Realizable k-ε Models 49
4.3.1. Standard k-ε Model 49
4.3.1.1. Overview 49
4.3.1.2. Transport Equations for the Standard k-ε Model 50
4.3.1.3. Modeling the Turbulent Viscosity 50
4.3.1.4. Model Constants 50
4.3.2. RNG k-ε Model 51
4.3.2.1. Overview 51
4.3.2.2. Transport Equations for the RNG k-ε Model 51
4.3.2.3. Modeling the Effective Viscosity 52
4.3.2.4. RNG Swirl Modification 52
4.3.2.5. Calculating the Inverse Effective Prandtl Numbers 53
4.3.2.6. The R-ε Term in the ε Equation 53
4.3.2.7. Model Constants 53
4.3.3. Realizable k-ε Model 54
4.3.3.1. Overview 54
4.3.3.2. Transport Equations for the Realizable k-ε Model 55

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

4.3.3.3. Modeling the Turbulent Viscosity 56
4.3.3.4. Model Constants 57
4.3.4. Modeling Turbulent Production in the k-ε Models 57
4.3.5. Remarks on the Linearization of Destruction Terms 57
4.3.6. Effects of Buoyancy on Turbulence in the k-ε Models 58
4.3.7. Turbulence Damping 59
4.3.8. Effects of Compressibility on Turbulence in the k-ε Models 59
4.3.9. Convective Heat and Mass Transfer Modeling in the k-ε Models 59
4.4. Standard, BSL, and SST k-ω Models 60
4.4.1. Standard k-ω Model 61
4.4.1.1. Overview 61
4.4.1.2. Transport Equations for the Standard k-ω Model 62
4.4.1.3. Modeling the Effective Diffusivity 62
4.4.1.3.1. Low-Reynolds Number Correction 62
4.4.1.4. Modeling the Turbulence Production 63
4.4.1.4.1. Production of k 63
4.4.1.4.2. Production of ω 63
4.4.1.5. Modeling the Turbulence Dissipation 63
4.4.1.5.1. Dissipation of k 63
4.4.1.5.2. Dissipation of ω 64
4.4.1.5.3. Compressibility Effects 64
4.4.1.6. Model Constants 65
4.4.2. Baseline (BSL) k-ω Model 65
4.4.2.1. Overview 65
4.4.2.2. Transport Equations for the BSL k-ω Model 65
4.4.2.3. Modeling the Effective Diffusivity 66
4.4.2.4. Modeling the Turbulence Production 66
4.4.2.4.1. Production of k 66
4.4.2.4.2. Production of ω 66
4.4.2.5. Modeling the Turbulence Dissipation 67
4.4.2.5.1. Dissipation of k 67
4.4.2.5.2. Dissipation of ω 67
4.4.2.6. Cross-Diffusion Modification 67
4.4.2.7. Model Constants 68
4.4.3. Shear-Stress Transport (SST) k-ω Model 68
4.4.3.1. Overview 68
4.4.3.2. Modeling the Turbulent Viscosity 68
4.4.3.3. Model Constants 68
4.4.3.4. Treatment of the SST Model for Icing Simulations 69
4.4.4. Effects of Buoyancy on Turbulence in the k-ω Models 69
4.4.5. Turbulence Damping 70
4.5. Generalized k-ω (GEKO) Model 70
4.5.1. Model Formulation 71
4.5.2. Limitations 73
4.6. k-kl-ω Transition Model 73
4.6.1. Overview 73
4.6.2. Transport Equations for the k-kl-ω Model 73
4.6.2.1. Model Constants 76
4.7. Transition SST Model 77
4.7.1. Overview 77
4.7.2. Transport Equations for the Transition SST Model 78

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

4.7.2.1. Separation-Induced Transition Correction 79
4.7.2.2. Coupling the Transition Model and SST Transport Equations 80
4.7.2.3. Transition SST and Rough Walls 80
4.7.3. Mesh Requirements 81
4.7.4. Specifying Inlet Turbulence Levels 84
4.8. Intermittency Transition Model 85
4.8.1. Overview 85
4.8.2. Transport Equations for the Intermittency Transition Model 86
4.8.3. Coupling with the Other Models 90
4.8.4. Intermittency Transition Model and Rough Walls 90
4.9. Algebraic Transition Model 90
4.9.1. Overview 90
4.9.2. Model Formulation 91
4.9.3. Transition Correlation 92
4.9.4. Pressure Gradient Function 92
4.9.5. Bubble Separation 92
4.9.6. Coefficients for the Algebraic Transition Model 93
4.9.7. Algebraic Transition Model and Rough Walls 94
4.10. Explicit Algebraic Reynolds Stress Model (EARSM) 94
4.11. Reynolds Stress Model (RSM) 96
4.11.1. Overview 97
4.11.2. Reynolds Stress Transport Equations 98
4.11.3. Modeling Turbulent Diffusive Transport 98
4.11.4. Modeling the Pressure-Strain Term 99
4.11.4.1. Linear Pressure-Strain Model 99
4.11.4.2. Low-Re Modifications to the Linear Pressure-Strain Model 100
4.11.4.3. Quadratic Pressure-Strain Model 101
4.11.4.4. Stress-Omega Model 101
4.11.4.5. Stress-BSL Model 103
4.11.5. Effects of Buoyancy on Turbulence 103
4.11.6. Modeling the Turbulence Kinetic Energy 103
4.11.7. Modeling the Dissipation Rate 104
4.11.8. Modeling the Turbulent Viscosity 104
4.11.9. Wall Boundary Conditions 105
4.11.10. Convective Heat and Mass Transfer Modeling 105
4.12. Scale-Adaptive Simulation (SAS) Model 106
4.12.1. Overview 106
4.12.2. Transport Equations for the SST-SAS Model 107
4.12.3. SAS with Other ω-Based Turbulence Models 109
4.13. Detached Eddy Simulation (DES) 109
4.13.1. Overview 109
4.13.2. DES with the Spalart-Allmaras Model 110
4.13.3. DES with the Realizable k-ε Model 111
4.13.4. DES with the BSL or SST k-ω Model 111
4.13.5. DES with the Transition SST Model 112
4.13.6. Improved Delayed Detached Eddy Simulation (IDDES) 113
4.13.6.1. Overview of IDDES 113
4.13.6.2. IDDES Model Formulation 113
4.14. Shielded Detached Eddy Simulation (SDES) 114
4.14.1. Shielding Function 114
4.14.2. LES Mode of SDES 116

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

4.15. Stress-Blended Eddy Simulation (SBES) 117
4.15.1. Stress Blending 117
4.15.2. SDES and SBES Example 118
4.16. Large Eddy Simulation (LES) Model 119
4.16.1. Overview 119
4.16.2. Subgrid-Scale Models 120
4.16.2.1. Smagorinsky-Lilly Model 121
4.16.2.2. Dynamic Smagorinsky-Lilly Model 122
4.16.2.3. Wall-Adapting Local Eddy-Viscosity (WALE) Model 123
4.16.2.4. Algebraic Wall-Modeled LES Model (WMLES) 123
4.16.2.4.1. Algebraic WMLES Model Formulation 124
4.16.2.4.1.1. Reynolds Number Scaling 124
4.16.2.4.2. Algebraic WMLES S-Omega Model Formulation 125
4.16.2.5. Dynamic Kinetic Energy Subgrid-Scale Model 126
4.16.3. Inlet Boundary Conditions for Scale Resolving Simulations 126
4.16.3.1. Vortex Method 127
4.16.3.2. Spectral Synthesizer 129
4.16.3.3. Synthetic Turbulence Generator 129
4.16.3.3.1. Limitations 131
4.17. Embedded Large Eddy Simulation (ELES) 132
4.17.1. Overview 132
4.17.2. Selecting a Model 133
4.17.3. Interfaces Treatment 133
4.17.3.1. RANS-LES Interface 133
4.17.3.2. LES-RANS Interface 134
4.17.3.3. Internal Interface Without LES Zone 134
4.17.3.4. Grid Generation Guidelines 135
4.18. Near-Wall Treatments for Wall-Bounded Turbulent Flows 135
4.18.1. Overview 136
4.18.2. Wall Treatment for ε-based Models 138
4.18.2.1. Standard Wall Functions 138
4.18.2.1.1. Momentum 138
4.18.2.1.2. Energy 139
4.18.2.1.3. Species 141
4.18.2.1.4. Turbulence 142
4.18.2.2. Scalable Wall Functions 143
4.18.2.3. Non-Equilibrium Wall Functions 143
4.18.2.3.1. Standard Wall Functions vs. Non-Equilibrium Wall Functions 144
4.18.2.3.2. Limitations of the Wall Function Approach 145
4.18.2.4. Enhanced Wall Treatment ε-Equation (EWT-ε) 145
4.18.2.4.1. Two-Layer Model for Enhanced Wall Treatment 145
4.18.2.4.2. Enhanced Wall Treatment for Momentum and Energy Equations 147
4.18.2.5. Menter-Lechner ε-Equation (ML-ε) 149
4.18.2.5.1. Momentum Equations 151
4.18.2.5.2. k-ε Turbulence Models 151
4.18.2.5.3. Iteration Improvements 151
4.18.2.6. User-Defined Wall Functions 151
4.18.3. y+-Insensitive Near-Wall Treatment for ω-based Turbulence Models 152
4.18.3.1. Momentum Equations 153
4.18.3.2. Turbulence 153
4.18.3.3. Energy 156

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

4.18.3.4. Limitations 156
4.18.4. LES Near-Wall Treatment 156
4.18.4.1. Kader Blending Wall Functions 156
4.18.4.2. Harmonic Blending Wall Functions Based on r+ 157
4.18.4.3. Werner and Wengle Wall Treatment 157
4.18.5. Wall Roughness Effects in Turbulent Wall-Bounded Flows 158
4.19. Curvature Correction for the Spalart-Allmaras and Two-Equation Models 162
4.20. Corner Flow Correction 164
4.21. Production Limiters for Two-Equation Models 165
4.22. Turbulence Damping 167
4.23. Definition of Turbulence Scales 168
4.23.1. RANS and Hybrid (SAS, DES, and SDES) Turbulence Models 168
4.23.2. Large Eddy Simulation (LES) Models 169
4.23.3. Stress-Blended Eddy Simulation (SBES) Model 169
5. Heat Transfer 171
5.1. Introduction 171
5.2. Modeling Conductive and Convective Heat Transfer 171
5.2.1. Heat Transfer Theory 172
5.2.1.1. The Energy Equation 172
5.2.1.2. The Energy Equation in Moving Reference Frames 173
5.2.1.3. The Energy Equation for the Non-Premixed Combustion Model 173
5.2.1.4. Inclusion of Pressure Work and Kinetic Energy Terms 173
5.2.1.5. Inclusion of the Viscous Dissipation Terms 173
5.2.1.6. Inclusion of the Species Diffusion Term 174
5.2.1.7. Energy Sources Due to Reaction 174
5.2.1.8. Energy Sources Due To Radiation 175
5.2.1.9. Energy Source Due To Joule Heating 175
5.2.1.10. Interphase Energy Sources 175
5.2.1.11. Energy Equation in Solid Regions 175
5.2.1.12. Anisotropic Conductivity in Solids 175
5.2.1.13. Diffusion at Inlets 176
5.2.2. Natural Convection and Buoyancy-Driven Flows Theory 176
5.2.3. The Two-Temperature Model for Hypersonic Flows 177
5.2.3.1. Energy Equations 178
5.2.3.2. Equation of States 179
5.2.3.3. Specific Heat 180
5.2.3.3.1. NASA-9-Piecewise-Polynomial Functions for the Two-Temperature Model 180
5.2.3.3.2. Boltzmann Energy Distribution Theory 180
5.2.3.4. Transport Properties 181
5.2.3.4.1. Blottner/Eucken/Constant Lewis 181
5.2.3.4.2. Gupta's Mixing Law 182
5.2.3.5. Limitations 183
5.3. Modeling Radiation 183
5.3.1. Overview and Limitations 184
5.3.1.1. Advantages and Limitations of the DTRM 185
5.3.1.2. Advantages and Limitations of the P-1 Model 185
5.3.1.3. Advantages and Limitations of the Rosseland Model 185
5.3.1.4. Advantages and Limitations of the DO Model 186
5.3.1.5. Advantages and Limitations of the S2S Model 186
5.3.1.6. Advantages and Limitations of the MC Model 187
5.3.2. Radiative Transfer Equation 189

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

5.3.3. P-1 Radiation Model Theory 190
5.3.3.1. The P-1 Model Equations 190
5.3.3.2. Anisotropic Scattering 191
5.3.3.3. Particulate Effects in the P-1 Model 192
5.3.3.4. Boundary Condition Treatment for the P-1 Model at Walls 193
5.3.3.5. Boundary Condition Treatment for the P-1 Model at Flow Inlets and Exits 193
5.3.4. Rosseland Radiation Model Theory 194
5.3.4.1. The Rosseland Model Equations 194
5.3.4.2. Anisotropic Scattering 194
5.3.4.3. Boundary Condition Treatment at Walls 194
5.3.4.4. Boundary Condition Treatment at Flow Inlets and Exits 195
5.3.5. Discrete Transfer Radiation Model (DTRM) Theory 195
5.3.5.1. The DTRM Equations 195
5.3.5.2. Ray Tracing 196
5.3.5.3. Clustering 196
5.3.5.4. Boundary Condition Treatment for the DTRM at Walls 197
5.3.5.5. Boundary Condition Treatment for the DTRM at Flow Inlets and outlets 197
5.3.6. Discrete Ordinates (DO) Radiation Model Theory 198
5.3.6.1. The DO Model Equations 198
5.3.6.2. Energy Coupling and the DO Model 199
5.3.6.2.1. Limitations of DO/Energy Coupling 200
5.3.6.3. Angular Discretization and Pixelation 200
5.3.6.4. Anisotropic Scattering 203
5.3.6.5. Particulate Effects in the DO Model 204
5.3.6.6. Boundary and Cell Zone Condition Treatment at Opaque Walls 204
5.3.6.6.1. Gray Diffuse Walls 206
5.3.6.6.2. Non-Gray Diffuse Walls 206
5.3.6.7. Cell Zone and Boundary Condition Treatment at Semi-Transparent Walls 207
5.3.6.7.1. Semi-Transparent Interior Walls 207
5.3.6.7.2. Specular Semi-Transparent Walls 209
5.3.6.7.3. Diffuse Semi-Transparent Walls 211
5.3.6.7.4. Partially Diffuse Semi-Transparent Walls 211
5.3.6.7.5. Semi-Transparent Exterior Walls 212
5.3.6.7.6. Limitations 214
5.3.6.7.7. Solid Semi-Transparent Media 214
5.3.6.8. Boundary Condition Treatment at Specular Walls and Symmetry Boundaries 214
5.3.6.9. Boundary Condition Treatment at Periodic Boundaries 214
5.3.6.10. Boundary Condition Treatment at Flow Inlets and Outlets 214
5.3.7. Surface-to-Surface (S2S) Radiation Model Theory 214
5.3.7.1. Gray-Diffuse Radiation 215
5.3.7.2. The S2S Model Equations 215
5.3.7.3. Clustering 216
5.3.7.3.1. Clustering and View Factors 216
5.3.7.3.2. Clustering and Radiosity 217
5.3.8. Monte Carlo (MC) Radiation Model Theory 217
5.3.8.1. The MC Model Equations 217
5.3.8.1.1. Monte Carlo Solution Accuracy 218
5.3.8.2. Boundary Condition Treatment for the MC Model 218
5.3.9. Radiation in Combusting Flows 218
5.3.9.1. The Weighted-Sum-of-Gray-Gases Model 218
5.3.9.1.1. When the Total (Static) Gas Pressure is Not Equal to 1 atm 220

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

5.3.9.2. The Effect of Soot on the Absorption Coefficient 220
5.3.9.3. The Effect of Particles on the Absorption Coefficient 221
5.3.10. Choosing a Radiation Model 221
5.3.10.1. External Radiation 222
6. Heat Exchangers 223
6.1. The Macro Heat Exchanger Models 223
6.1.1. Overview of the Macro Heat Exchanger Models 223
6.1.2. Restrictions of the Macro Heat Exchanger Models 225
6.1.3. Macro Heat Exchanger Model Theory 226
6.1.3.1. Streamwise Pressure Drop 227
6.1.3.2. Heat Transfer Effectiveness 228
6.1.3.3. Heat Rejection 229
6.1.3.4. Macro Heat Exchanger Group Connectivity 231
6.2. The Dual Cell Model 231
6.2.1. Overview of the Dual Cell Model 232
6.2.2. Restrictions of the Dual Cell Model 232
6.2.3. Dual Cell Model Theory 232
6.2.3.1. NTU Relations 233
6.2.3.2. Heat Rejection 234
7. Modelling with Finite-Rate Chemistry 237
7.1. Species Transport and Finite-Rate Chemistry 237
7.1.1. Volumetric Reactions 237
7.1.1.1. Species Transport Equations 238
7.1.1.1.1. Mass Diffusion in Laminar Flows 238
7.1.1.1.2. Mass Diffusion in Turbulent Flows 239
7.1.1.1.3. Full Multicomponent Diffusion 239
7.1.1.1.3.1. General Theory 239
7.1.1.1.3.2. Maxwell-Stefan Equations 239
7.1.1.1.4. Kinetic Theory Parameters for Diffusion Coefficients 241
7.1.1.1.5. Unity Lewis Number 241
7.1.1.1.6. Anisotropic Species Diffusion 242
7.1.1.1.7. Thermal Diffusion Coefficients 242
7.1.1.1.8. Treatment of Species Transport in the Energy Equation 242
7.1.1.1.9. Diffusion at Inlets 242
7.1.1.2. The Generalized Finite-Rate Formulation for Reaction Modeling 243
7.1.1.2.1. Direct Use of Finite-Rate Kinetics (no TCI) 243
7.1.1.2.2. Pressure-Dependent Reactions 246
7.1.1.2.3. The Eddy-Dissipation Model 247
7.1.1.2.4. The Eddy-Dissipation Model for LES 248
7.1.1.2.5. The Eddy-Dissipation-Concept (EDC) Model 249
7.1.1.2.5.1. The Standard EDC Model 249
7.1.1.2.5.2. The Partially Stirred Reactor EDC Model 250
7.1.1.2.6. The Thickened Flame Model 251
7.1.1.2.7. The Relaxation to Chemical Equilibrium Model 254
7.1.1.3. Finite-Rate Chemistry with the Two-Temperature Model 256
7.1.2. Wall Surface Reactions and Chemical Vapor Deposition 256
7.1.2.1. Surface Coverage Reaction Rate Modification 258
7.1.2.2. Reaction-Diffusion Balance for Surface Chemistry 258
7.1.2.3. Slip Boundary Formulation for Low-Pressure Gas Systems 259
7.1.3. Particle Reactions 261
7.1.3.1. Combusting Particle Surface Reactions 261

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

7.1.3.1.1. General Description 261
7.1.3.1.2. Ansys Fluent Model Formulation 262
7.1.3.1.3. Extension for Stoichiometries with Multiple Gas Phase Reactants 263
7.1.3.1.4. Solid-Solid Reactions 264
7.1.3.1.5. Solid Decomposition Reactions 264
7.1.3.1.6. Solid Deposition Reactions 264
7.1.3.1.7. Gaseous Solid Catalyzed Reactions on the Particle Surface 265
7.1.3.2. Multicomponent Particles with Chemical Reactions 265
7.1.4. Electrochemical Reactions 267
7.1.4.1. Overview 267
7.1.4.2. Electrochemical Reaction Model Theory 267
7.1.5. Reacting Channel Model 270
7.1.5.1. Overview 271
7.1.5.2. Reacting Channel Model Theory 271
7.1.5.2.1. Flow Inside the Reacting Channel 272
7.1.5.2.2. Surface Reactions in the Reacting Channel 273
7.1.5.2.3. Porous Medium Inside Reacting Channel 273
7.1.5.2.4. Outer Flow in the Shell 274
7.1.6. Reactor Network Model 274
7.1.6.1. Reactor Network Model Theory 275
7.1.6.1.1. Reactor network temperature solution 276
7.2. Composition PDF Transport 276
7.2.1. Overview 276
7.2.2. Composition PDF Transport Theory 276
7.2.3. The Lagrangian Solution Method 278
7.2.3.1. Particle Convection 278
7.2.3.2. Particle Mixing 279
7.2.3.2.1. The Modified Curl Model 279
7.2.3.2.2. The IEM Model 280
7.2.3.2.3. The EMST Model 280
7.2.3.2.4. Liquid Reactions 280
7.2.3.3. Particle Reaction 280
7.2.4. The Eulerian Solution Method 281
7.2.4.1. Reaction 282
7.2.4.2. Mixing 282
7.2.4.3. Correction 283
7.2.4.4. Calculation of Composition Mean and Variance 283
7.3. Chemistry Acceleration 283
7.3.1. Overview and Limitations 284
7.3.2. In-Situ Adaptive Tabulation (ISAT) 284
7.3.3. Dynamic Mechanism Reduction 286
7.3.3.1. Directed Relation Graph (DRG) Method for Mechanism Reduction 287
7.3.4. Chemistry Agglomeration 288
7.3.4.1. Binning Algorithm 290
7.3.5. Chemical Mechanism Dimension Reduction 291
7.3.5.1. Selecting the Represented Species 292
7.3.6. Dynamic Cell Clustering with Ansys Fluent CHEMKIN-CFD Solver 292
7.3.7. Dynamic Adaptive Chemistry with Ansys Fluent CHEMKIN-CFD Solver 293
8. Modelling of Turbulent Combustion With Reduced Order 295
8.1. Non-Premixed Combustion 295
8.1.1. Introduction 295

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

8.1.2. Non-Premixed Combustion and Mixture Fraction Theory 296
8.1.2.1. Mixture Fraction Theory 296
8.1.2.1.1. Definition of the Mixture Fraction 296
8.1.2.1.2. Transport Equations for the Mixture Fraction 298
8.1.2.1.3. The Non-Premixed Model for LES 299
8.1.2.1.4. The Non-Premixed Model with the SBES Turbulence Model 299
8.1.2.1.5. Mixture Fraction vs. Equivalence Ratio 300
8.1.2.1.6. Relationship of Mixture Fraction to Species Mass Fraction, Density, and Temperature 300
8.1.2.2. Modeling of Turbulence-Chemistry Interaction 301
8.1.2.2.1. Description of the Probability Density Function 301
8.1.2.2.2. Derivation of Mean Scalar Values from the Instantaneous Mixture Fraction 302
8.1.2.2.3. The Assumed-Shape PDF 303
8.1.2.2.3.1. The Double Delta Function PDF 303
8.1.2.2.3.2. The β-Function PDF 304
8.1.2.3. Non-Adiabatic Extensions of the Non-Premixed Model 305
8.1.2.4. Chemistry Tabulation 307
8.1.2.4.1. Look-Up Tables for Adiabatic Systems 307
8.1.2.4.2. 3D Look-Up Tables for Non-Adiabatic Systems 309
8.1.2.4.3. Generating Lookup Tables Through Automated Grid Refinement 312
8.1.3. Restrictions and Special Cases for Using the Non-Premixed Model 313
8.1.3.1. Restrictions on the Mixture Fraction Approach 313
8.1.3.2. Using the Non-Premixed Model for Liquid Fuel or Coal Combustion 316
8.1.3.3. Using the Non-Premixed Model with Flue Gas Recycle 317
8.1.3.4. Using the Non-Premixed Model with the Inert Model 318
8.1.3.4.1. Mixture Composition 318
8.1.3.4.1.1. Property Evaluation 319
8.1.4. The Diffusion Flamelet Models Theory 319
8.1.4.1. Restrictions and Assumptions 319
8.1.4.2. The Flamelet Concept 320
8.1.4.2.1. Overview 320
8.1.4.2.2. Strain Rate and Scalar Dissipation 321
8.1.4.2.3. Embedding Diffusion Flamelets in Turbulent Flames 322
8.1.4.3. Flamelet Generation 323
8.1.4.4. Flamelet Import 323
8.1.5. The Steady Diffusion Flamelet Model Theory 324
8.1.5.1. Overview 325
8.1.5.2. Multiple Steady Flamelet Libraries 326
8.1.5.3. Steady Diffusion Flamelet Automated Grid Refinement 326
8.1.5.4. Non-Adiabatic Steady Diffusion Flamelets 327
8.1.6. The Unsteady Diffusion Flamelet Model Theory 327
8.1.6.1. The Eulerian Unsteady Laminar Flamelet Model 328
8.1.6.1.1. Liquid Reactions 330
8.1.6.2. The Diesel Unsteady Laminar Flamelet Model 330
8.1.6.3. Multiple Diesel Unsteady Flamelets 331
8.1.6.4. Multiple Diesel Unsteady Flamelets with Flamelet Reset 332
8.1.6.4.1. Resetting the Flamelets 332
8.2. Premixed Combustion 332
8.2.1. Overview 333
8.2.2. C-Equation Model Theory 333
8.2.2.1. Propagation of the Flame Front 334

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

8.2.3. G-Equation Model Theory 335
8.2.3.1. Numerical Solution of the G-equation 336
8.2.4. Turbulent Flame Speed Models 336
8.2.4.1. Zimont Turbulent Flame Speed Closure Model 336
8.2.4.1.1. Zimont Turbulent Flame Speed Closure for LES 337
8.2.4.1.2. Flame Stretch Effect 338
8.2.4.1.3. Wall Damping 339
8.2.4.2. Peters Flame Speed Model 339
8.2.4.2.1. Peters Flame Speed Model for LES 341
8.2.5. Calculation of Properties 341
8.2.5.1. Calculation of Temperature 341
8.2.5.1.1. Adiabatic Temperature Calculation 341
8.2.5.1.2. Non-Adiabatic Temperature Calculation 341
8.2.5.2. Calculation of Density 342
8.2.5.3. Laminar Flame Speed 342
8.2.5.4. Unburnt Density and Thermal Diffusivity 343
8.3. Partially Premixed Combustion 343
8.3.1. Overview 343
8.3.2. Partially Premixed Combustion Theory 344
8.3.2.1. Chemical Equilibrium and Steady Diffusion Flamelet Models 344
8.3.2.2. Flamelet Generated Manifold (FGM) Model 345
8.3.2.2.1. Premixed FGMs in Reaction Progress Variable Space 345
8.3.2.2.2. Premixed FGMs in Physical Space 348
8.3.2.2.3. Diffusion FGMs 349
8.3.2.2.4. Nonadiabatic Flamelet Generated Manifold (FGM) 350
8.3.2.3. FGM Turbulent Closure 351
8.3.2.3.1. Scalar Transport with FGM Closure 352
8.3.2.4. Calculation of Mixture Properties 354
8.3.2.5. Calculation of Unburnt Properties 355
8.3.2.6. Laminar Flame Speed 355
8.3.2.7. Strained Laminar Flame Speed 357
8.3.2.7.1. Strained Non-Adiabatic Flame Speed 360
8.3.2.8. Generating PDF Lookup Tables Through Automated Grid Refinement 361
9. Pollutant Formation 363
9.1. NOx Formation 363
9.1.1. Overview 363
9.1.1.1. NOx Modeling in Ansys Fluent 363
9.1.1.2. NOx Formation and Reduction in Flames 364
9.1.2. Governing Equations for NOx Transport 364
9.1.3. Thermal NOx Formation 365
9.1.3.1. Thermal NOx Reaction Rates 365
9.1.3.2. The Quasi-Steady Assumption for [N] 366
9.1.3.3. Thermal NOx Temperature Sensitivity 366
9.1.3.4. Decoupled Thermal NOx Calculations 366
9.1.3.5. Approaches for Determining O Radical Concentration 366
9.1.3.5.1. Method 1: Equilibrium Approach 367
9.1.3.5.2. Method 2: Partial Equilibrium Approach 367
9.1.3.5.3. Method 3: Predicted O Approach 367
9.1.3.6. Approaches for Determining OH Radical Concentration 368
9.1.3.6.1. Method 1: Exclusion of OH Approach 368
9.1.3.6.2. Method 2: Partial Equilibrium Approach 368

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

9.1.3.6.3. Method 3: Predicted OH Approach 368
9.1.3.7. Summary 368
9.1.4. Prompt NOx Formation 368
9.1.4.1. Prompt NOx Combustion Environments 369
9.1.4.2. Prompt NOx Mechanism 369
9.1.4.3. Prompt NOx Formation Factors 369
9.1.4.4. Primary Reaction 369
9.1.4.5. Modeling Strategy 370
9.1.4.6. Rate for Most Hydrocarbon Fuels 370
9.1.4.7. Oxygen Reaction Order 371
9.1.5. Fuel NOx Formation 371
9.1.5.1. Fuel-Bound Nitrogen 371
9.1.5.2. Reaction Pathways 371
9.1.5.3. Fuel NOx from Gaseous and Liquid Fuels 372
9.1.5.3.1. Fuel NOx from Intermediate Hydrogen Cyanide (HCN) 372
9.1.5.3.1.1. HCN Production in a Gaseous Fuel 373
9.1.5.3.1.2. HCN Production in a Liquid Fuel 373
9.1.5.3.1.3. HCN Consumption 373
9.1.5.3.1.4. HCN Sources in the Transport Equation 374
9.1.5.3.1.5. NOx Sources in the Transport Equation 374
9.1.5.3.2. Fuel NOx from Intermediate Ammonia (NH3) 374
9.1.5.3.2.1. NH3 Production in a Gaseous Fuel 375
9.1.5.3.2.2. NH3 Production in a Liquid Fuel 375
9.1.5.3.2.3. NH3 Consumption 376
9.1.5.3.2.4. NH3 Sources in the Transport Equation 376
9.1.5.3.2.5. NOx Sources in the Transport Equation 376
9.1.5.3.3. Fuel NOx from Coal 377
9.1.5.3.3.1. Nitrogen in Char and in Volatiles 377
9.1.5.3.3.2. Coal Fuel NOx Scheme A 377
9.1.5.3.3.3. Coal Fuel NOx Scheme B 377
9.1.5.3.3.4. HCN Scheme Selection 378
9.1.5.3.3.5. NOx Reduction on Char Surface 378
9.1.5.3.3.5.1. BET Surface Area 379
9.1.5.3.3.5.2. HCN from Volatiles 379
9.1.5.3.3.6. Coal Fuel NOx Scheme C 379
9.1.5.3.3.7. Coal Fuel NOx Scheme D 380
9.1.5.3.3.8. NH3 Scheme Selection 381
9.1.5.3.3.8.1. NH3 from Volatiles 381
9.1.5.3.4. Fuel Nitrogen Partitioning for HCN and NH3 Intermediates 381
9.1.6. NOx Formation from Intermediate N2O 382
9.1.6.1. N2O - Intermediate NOx Mechanism 382
9.1.7. NOx Reduction by Reburning 383
9.1.7.1. Instantaneous Approach 383
9.1.7.2. Partial Equilibrium Approach 384
9.1.7.2.1. NOx Reduction Mechanism 385
9.1.8. NOx Reduction by SNCR 386
9.1.8.1. Ammonia Injection 386
9.1.8.2. Urea Injection 388
9.1.8.3. Transport Equations for Urea, HNCO, and NCO 389
9.1.8.4. Urea Production due to Reagent Injection 390
9.1.8.5. NH3 Production due to Reagent Injection 390

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

9.1.8.6. HNCO Production due to Reagent Injection 390
9.1.9. NOx Formation in Turbulent Flows 390
9.1.9.1. The Turbulence-Chemistry Interaction Model 391
9.1.9.2. The PDF Approach 391
9.1.9.3. The General Expression for the Mean Reaction Rate 391
9.1.9.4. The Mean Reaction Rate Used in Ansys Fluent 392
9.1.9.5. Statistical Independence 392
9.1.9.6. The Beta PDF Option 392
9.1.9.7. The Gaussian PDF Option 393
9.1.9.8. The Calculation Method for the Variance 393
9.2. Soot Formation 394
9.2.1. Overview and Limitations 394
9.2.1.1. Predicting Soot Formation 394
9.2.1.2. Restrictions on Soot Modeling 395
9.2.2. Soot Model Theory 395
9.2.2.1. The One-Step Soot Formation Model 395
9.2.2.2. The Two-Step Soot Formation Model 396
9.2.2.2.1. Soot Generation Rate 396
9.2.2.2.2. Nuclei Generation Rate 397
9.2.2.3. The Moss-Brookes Model 398
9.2.2.3.1. The Moss-Brookes-Hall Model 399
9.2.2.3.2. Soot Formation in Turbulent Flows 400
9.2.2.3.2.1. The Turbulence-Chemistry Interaction Model 401
9.2.2.3.2.2. The PDF Approach 401
9.2.2.3.2.3. The Mean Reaction Rate 401
9.2.2.3.2.4. The PDF Options 401
9.2.2.3.3. The Effect of Soot on the Radiation Absorption Coefficient 401
9.2.2.4. The Method of Moments Model 402
9.2.2.4.1. Soot Particle Population Balance 402
9.2.2.4.2. Moment Transport Equations 403
9.2.2.4.3. Nucleation 404
9.2.2.4.4. Coagulation 406
9.2.2.4.5. Surface Growth and Oxidation 409
9.2.2.4.6. Soot Aggregation 412
9.3. Decoupled Detailed Chemistry Model 416
9.3.1. Overview 416
9.3.1.1. Limitations 417
9.3.2. Decoupled Detailed Chemistry Model Theory 417
10. Engine Ignition 419
10.1. Spark Model 419
10.1.1. Overview and Limitations 419
10.1.2. Spark Model Theory 419
10.2. Autoignition Models 422
10.2.1. Model Overview 422
10.2.2. Model Limitations 423
10.2.3. Ignition Model Theory 423
10.2.3.1. Transport of Ignition Species 423
10.2.3.2. Knock Modeling 424
10.2.3.2.1. Modeling of the Source Term 424
10.2.3.2.2. Correlations 425
10.2.3.2.3. Energy Release 425

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

10.2.3.3. Ignition Delay Modeling 426
10.2.3.3.1. Modeling of the Source Term 426
10.2.3.3.2. Correlations 426
10.2.3.3.3. Energy Release 427
10.3. Crevice Model 427
10.3.1. Overview 427
10.3.1.1. Model Parameters 428
10.3.2. Limitations 429
10.3.3. Crevice Model Theory 430
11. Aerodynamically Generated Noise 431
11.1. Overview 431
11.1.1. Direct Method 431
11.1.2. Integral Method by Ffowcs Williams and Hawkings 432
11.1.3. Method Based on Wave Equation 433
11.1.4. Broadband Noise Source Models 433
11.2. Acoustics Model Theory 434
11.2.1. The Ffowcs Williams and Hawkings Model 434
11.2.2. Wave Equation Model 437
11.2.2.1. Limitations 437
11.2.2.2. Governing Equations and Boundary Conditions 437
11.2.2.3. Method of Numerical Solution 438
11.2.2.4. Preventing Non-Physical Reflections of Sound Waves 438
11.2.2.4.1. Mesh Quality 438
11.2.2.4.2. Filtering of the Sound Source Term 439
11.2.2.4.3. Ramping in Time and Limiting in Space (Masking) of the Sound Source Term 439
11.2.2.4.4. Damping of Solution in a Sponge Region Using Artificial Viscosity 439
11.2.2.5. Kirchhoff Integral 440
11.2.2.5.1. Compatibility and Limitations 440
11.2.2.5.2. Mathematical Formulation 441
11.2.3. Broadband Noise Source Models 441
11.2.3.1. Proudman's Formula 441
11.2.3.2. The Jet Noise Source Model 442
11.2.3.3. The Boundary Layer Noise Source Model 443
11.2.3.4. Source Terms in the Linearized Euler Equations 444
11.2.3.5. Source Terms in Lilley's Equation 445
12. Discrete Phase 447
12.1. Introduction 447
12.1.1. The Euler-Lagrange Approach 447
12.2. Particle Motion Theory 448
12.2.1. Equations of Motion for Particles 448
12.2.1.1. Particle Force Balance 448
12.2.1.2. Particle Torque Balance 449
12.2.1.3. Inclusion of the Gravity Term 449
12.2.1.4. Other Forces 449
12.2.1.5. Forces in Moving Reference Frames 450
12.2.1.6. Thermophoretic Force 450
12.2.1.7. Brownian Force 451
12.2.1.8. Saffman's Lift Force 451
12.2.1.9. Magnus Lift Force 451
12.2.2. Turbulent Dispersion of Particles 452
12.2.2.1. Stochastic Tracking 453

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

12.2.2.1.1. The Integral Time 453
12.2.2.1.2. The Discrete Random Walk Model 454
12.2.2.1.3. Using the DRW Model 455
12.2.3. Integration of Particle Equation of Motion 455
12.3. Laws for Drag Coefficients 457
12.3.1. Spherical Drag Law 458
12.3.2. Non-spherical Drag Law 458
12.3.3. Stokes-Cunningham Drag Law 458
12.3.4. High-Mach-Number Drag Law 459
12.3.5. Dynamic Drag Model Theory 459
12.3.6. Dense Discrete Phase Model Drag Laws 460
12.3.7. Bubbly Flow Drag Laws 460
12.3.7.1. Ishii-Zuber Drag Model 460
12.3.7.2. Grace Drag Model 461
12.3.8. Rotational Drag Law 462
12.4. Laws for Heat and Mass Exchange 462
12.4.1. Inert Heating or Cooling (Law 1/Law 6) 463
12.4.2. Droplet Vaporization (Law 2) 465
12.4.2.1. Mass Transfer During Law 2—Diffusion Controlled Model 466
12.4.2.2. Mass Transfer During Law 2—Convection/Diffusion Controlled Model 467
12.4.2.3. Mass Transfer During Law 2—Thermolysis 468
12.4.2.4. Defining the Saturation Vapor Pressure and Diffusion Coefficient (or Binary Diffusivity) 468
12.4.2.5. Defining the Boiling Point and Latent Heat 469
12.4.2.6. Heat Transfer to the Droplet 470
12.4.3. Droplet Boiling (Law 3) 472
12.4.4. Devolatilization (Law 4) 473
12.4.4.1. Choosing the Devolatilization Model 474
12.4.4.2. The Constant Rate Devolatilization Model 474
12.4.4.3. The Single Kinetic Rate Model 474
12.4.4.4. The Two Competing Rates (Kobayashi) Model 475
12.4.4.5. The CPD Model 476
12.4.4.5.1. General Description 476
12.4.4.5.2. Reaction Rates 477
12.4.4.5.3. Mass Conservation 477
12.4.4.5.4. Fractional Change in the Coal Mass 478
12.4.4.5.5. CPD Inputs 479
12.4.4.6. Particle Swelling During Devolatilization 480
12.4.4.7. Heat Transfer to the Particle During Devolatilization 480
12.4.5. Surface Combustion (Law 5) 481
12.4.5.1. The Diffusion-Limited Surface Reaction Rate Model 482
12.4.5.2. The Kinetic/Diffusion Surface Reaction Rate Model 482
12.4.5.3. The Intrinsic Model 483
12.4.5.4. The Multiple Surface Reactions Model 484
12.4.5.4.1. Limitations 485
12.4.5.5. Heat and Mass Transfer During Char Combustion 485
12.4.6. Multicomponent Particle Definition (Law 7) 485
12.4.6.1. Raoult's Law 488
12.4.6.2. Peng-Robinson Real Gas Model 488
12.5. Vapor Liquid Equilibrium Theory 488
12.6. Physical Property Averaging 490

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

12.7. Wall-Particle Reflection Model Theory 492
12.7.1. Rough Wall Model 495
12.8. Wall-Jet Model Theory 496
12.9. Lagrangian Wall-Film Model Theory 497
12.9.1. Introduction 497
12.9.2. Leidenfrost Temperature Considerations 499
12.9.2.1. Default Wall Temperature Limiter 499
12.9.2.2. Leidenfrost Temperature Reporting 500
12.9.3. Interaction During Impact with a Boundary 500
12.9.3.1. The Stanton-Rutland Model 501
12.9.3.1.1. Regime Definition 501
12.9.3.1.2. Rebound 503
12.9.3.1.3. Splashing 503
12.9.3.2. The Kuhnke Model 508
12.9.3.2.1. Regime definition 508
12.9.3.2.2. Rebound 512
12.9.3.2.3. Splashing 512
12.9.3.3. The Stochastic Kuhnke Model 514
12.9.4. Separation and Stripping Submodels 517
12.9.5. Conservation Equations for Wall-Film Particles 517
12.9.5.1. Momentum 517
12.9.5.2. Mass Transfer from the Film 518
12.9.5.2.1. Film Vaporization and Boiling 518
12.9.5.2.1.1. Diffusion Controlled Model 519
12.9.5.2.1.2. Convection/Diffusion Controlled Model 520
12.9.5.2.1.3. Thermolysis Model 520
12.9.5.2.1.4. Gas-Side Boundary Layer Model 520
12.9.5.2.1.5. Film Boiling Model with Conduction Heat Transfer Model 520
12.9.5.2.1.6. Film Boiling Model with Convective Heat Transfer Model 521
12.9.5.2.2. Film Condensation 522
12.9.5.2.3. Multicomponent Film Models 524
12.9.5.3. Energy Transfer from the Film 524
12.9.5.3.1. Conduction Heat Transfer Model 524
12.9.5.3.2. Convective Heat Transfer Model 526
12.10. Wall Erosion 533
12.10.1. Finnie Erosion Model 534
12.10.2. Oka Erosion Model 534
12.10.3. McLaury Erosion Model 535
12.10.4. DNV Erosion Model 536
12.10.5. Modeling Erosion Rates in Dense Flows 537
12.10.5.1. Abrasive Erosion Caused by Solid Particles 537
12.10.5.2. Wall Shielding Effect in Dense Flow Regimes 538
12.10.6. Accretion 538
12.11. Particle–Wall Impingement Heat Transfer Theory 540
12.12. Atomizer Model Theory 541
12.12.1. The Plain-Orifice Atomizer Model 542
12.12.1.1. Internal Nozzle State 543
12.12.1.2. Coefficient of Discharge 544
12.12.1.3. Exit Velocity 546
12.12.1.4. Spray Angle 546
12.12.1.5. Droplet Diameter Distribution 546

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

12.12.2. The Pressure-Swirl Atomizer Model 548
12.12.2.1. Film Formation 549
12.12.2.2. Sheet Breakup and Atomization 550
12.12.3. The Air-Blast/Air-Assist Atomizer Model 552
12.12.4. The Flat-Fan Atomizer Model 553
12.12.5. The Effervescent Atomizer Model 554
12.13. Secondary Breakup Model Theory 554
12.13.1. Taylor Analogy Breakup (TAB) Model 555
12.13.1.1. Introduction 555
12.13.1.2. Use and Limitations 556
12.13.1.3. Droplet Distortion 556
12.13.1.4. Size of Child Droplets 557
12.13.1.5. Velocity of Child Droplets 558
12.13.1.6. Droplet Breakup 558
12.13.2. Wave Breakup Model 559
12.13.2.1. Introduction 559
12.13.2.2. Use and Limitations 560
12.13.2.3. Jet Stability Analysis 560
12.13.2.4. Droplet Breakup 561
12.13.3. KHRT Breakup Model 562
12.13.3.1. Introduction 562
12.13.3.2. Use and Limitations 562
12.13.3.3. Liquid Core Length 562
12.13.3.4. Rayleigh-Taylor Breakup 563
12.13.3.5. Droplet Breakup Within the Liquid Core 564
12.13.3.6. Droplet Breakup Outside the Liquid Core 564
12.13.4. Stochastic Secondary Droplet (SSD) Model 564
12.13.5. Madabhushi Breakup Model 565
12.13.6. Schmehl Breakup Model 569
12.14. Collision and Droplet Coalescence Model Theory 573
12.14.1. Introduction 573
12.14.2. Use and Limitations 574
12.14.3. Theory 574
12.14.3.1. Probability of Collision 574
12.14.3.2. Collision Outcomes 575
12.14.3.2.1. Bouncing 576
12.14.3.2.2. Bouncing and Coalescence 576
12.15. Discrete Element Method Collision Model 577
12.15.1. Theory 577
12.15.1.1. The Spring Collision Law 578
12.15.1.2. The Spring-Dashpot Collision Law 578
12.15.1.3. The Hertzian Collision Law 579
12.15.1.4. The Hertzian-Dashpot Collision Law 579
12.15.1.5. The Friction Collision Law 580
12.15.1.6. Rolling Friction Collision Law for DEM 581
12.15.1.7. DEM Parcels 581
12.15.1.8. Cartesian Collision Mesh 582
12.16. One-Way and Two-Way Coupling 582
12.16.1. Coupling Between the Discrete and Continuous Phases 583
12.16.2. Momentum Exchange 583
12.16.3. Heat Exchange 584

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

12.16.4. Mass Exchange 585
12.16.5. Under-Relaxation of the Interphase Exchange Terms 585
12.16.6. Interphase Exchange During Stochastic Tracking 586
12.17. Node Based Averaging 586
12.18. Blockage Effect 587
13. Modeling Macroscopic Particles 589
13.1. Momentum Transfer to Fluid Flow 589
13.2. Fluid Forces and Torques on Particle 590
13.3. Particle/Particle and Particle/Wall Collisions 591
13.4. Field Forces 592
13.5. Particle Deposition and Buildup 593
14. Multiphase Flows 595
14.1. Introduction 595
14.1.1. Multiphase Flow Regimes 595
14.1.1.1. Gas-Liquid or Liquid-Liquid Flows 595
14.1.1.2. Gas-Solid Flows 596
14.1.1.3. Liquid-Solid Flows 596
14.1.1.4. Three-Phase Flows 596
14.1.2. Examples of Multiphase Systems 597
14.2. Choosing a General Multiphase Model 598
14.2.1. Approaches to Multiphase Modeling 598
14.2.1.1. The Euler-Euler Approach 598
14.2.1.1.1. The VOF Model 599
14.2.1.1.2. The Mixture Model 599
14.2.1.1.3. The Eulerian Model 599
14.2.2. Model Comparisons 599
14.2.2.1. Detailed Guidelines 600
14.2.2.1.1. The Effect of Particulate Loading 601
14.2.2.1.2. The Significance of the Stokes Number 602
14.2.2.1.2.1. Examples 602
14.2.2.1.3. Other Considerations 602
14.2.3. Time Schemes in Multiphase Flow 602
14.2.4. Stability and Convergence 603
14.3. Volume of Fluid (VOF) Model Theory 604
14.3.1. Overview of the VOF Model 605
14.3.2. Limitations of the VOF Model 605
14.3.3. Steady-State and Transient VOF Calculations 605
14.3.4. Volume Fraction Equation 606
14.3.4.1. The Implicit Formulation 606
14.3.4.2. The Explicit Formulation 607
14.3.4.3. Interpolation Near the Interface 608
14.3.4.3.1. The Geometric Reconstruction Scheme 609
14.3.4.3.2. The Donor-Acceptor Scheme 609
14.3.4.3.3. The Compressive Interface Capturing Scheme for Arbitrary Meshes (CICSAM) 610
14.3.4.3.4. The Compressive Scheme and Interface-Model-based Variants 610
14.3.4.3.5. Bounded Gradient Maximization (BGM) 611
14.3.5. Material Properties 611
14.3.6. Momentum Equation 611
14.3.7. Energy Equation 612
14.3.8. Additional Scalar Equations 612
14.3.9. Surface Tension and Adhesion 613

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

14.3.9.1. Surface Tension 613
14.3.9.1.1. The Continuum Surface Force Model 613
14.3.9.1.2. The Continuum Surface Stress Model 614
14.3.9.1.3. Comparing the CSS and CSF Methods 615
14.3.9.1.4. When Surface Tension Effects Are Important 615
14.3.9.2. Wall Adhesion 616
14.3.9.3. Jump Adhesion 616
14.3.10. Open Channel Flow 617
14.3.10.1. Upstream Boundary Conditions 618
14.3.10.1.1. Pressure Inlet 618
14.3.10.1.2. Mass Flow Rate 618
14.3.10.1.3. Volume Fraction Specification 619
14.3.10.2. Downstream Boundary Conditions 619
14.3.10.2.1. Pressure Outlet 619
14.3.10.2.2. Outflow Boundary 619
14.3.10.2.3. Backflow Volume Fraction Specification 620
14.3.10.3. Numerical Beach Treatment 620
14.3.11. Open Channel Wave Boundary Conditions 621
14.3.11.1. Airy Wave Theory 622
14.3.11.2. Stokes Wave Theories 623
14.3.11.3. Cnoidal/Solitary Wave Theory 624
14.3.11.4. Choosing a Wave Theory 625
14.3.11.5. Superposition of Waves 628
14.3.11.6. Modeling of Random Waves Using Wave Spectrum 628
14.3.11.6.1. Definitions 629
14.3.11.6.2. Wave Spectrum Implementation Theory 629
14.3.11.6.2.1. Long-Crested Random Waves (Unidirectional) 629
14.3.11.6.2.1.1. Pierson-Moskowitz Spectrum 629
14.3.11.6.2.1.2. JONSWAP Spectrum 630
14.3.11.6.2.1.3. TMA Spectrum 630
14.3.11.6.2.2. Short-Crested Random Waves (Multi-Directional) 630
14.3.11.6.2.2.1. Cosine-2s Power Function (Frequency Independent) 631
14.3.11.6.2.2.2. Hyperbolic Function (Frequency Dependent) 631
14.3.11.6.2.3. Superposition of Individual Wave Components Using the Wave Spectrum 632
14.3.11.6.3. Choosing a Wave Spectrum and Inputs 633
14.3.11.7. Nomenclature for Open Channel Waves 635
14.3.12. Coupled Level-Set and VOF Model 636
14.3.12.1. Theory 637
14.3.12.1.1. Surface Tension Force 637
14.3.12.1.2. Re-initialization of the Level-set Function via the Geometrical Method 638
14.3.12.2. Limitations 640
14.4. Mixture Model Theory 640
14.4.1. Overview 641
14.4.2. Limitations of the Mixture Model 641
14.4.3. Continuity Equation 642
14.4.4. Momentum Equation 642
14.4.5. Energy Equation 643
14.4.6. Relative (Slip) Velocity and the Drift Velocity 643
14.4.7. Volume Fraction Equation for the Secondary Phases 645
14.4.8. Granular Properties 645

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

14.4.8.1. Collisional Viscosity 645
14.4.8.2. Kinetic Viscosity 645
14.4.9. Granular Temperature 646
14.4.10. Solids Pressure 647
14.4.11. Interfacial Area Concentration 647
14.4.11.1. Transport Equation Based Models 647
14.4.11.1.1. Hibiki-Ishii Model 648
14.4.11.1.2. Ishii-Kim Model 649
14.4.11.1.3. Yao-Morel Model 650
14.4.11.2. Algebraic Models 651
14.4.12. Flow Regime Modeling 652
14.4.12.1. Phase Pair Interaction for Hybrid Morphology 655
14.4.12.2. Algebraic Interfacial Area Density Method for Flow Regime Blending 657
14.5. Eulerian Model Theory 659
14.5.1. Overview of the Eulerian Model 660
14.5.2. Limitations of the Eulerian Model 661
14.5.3. Volume Fraction Equation 661
14.5.4. Conservation Equations 662
14.5.4.1. Equations in General Form 662
14.5.4.1.1. Conservation of Mass 662
14.5.4.1.2. Conservation of Momentum 662
14.5.4.1.3. Conservation of Energy 663
14.5.4.2. Equations Solved by Ansys Fluent 664
14.5.4.2.1. Continuity Equation 664
14.5.4.2.2. Fluid-Fluid Momentum Equations 664
14.5.4.2.3. Fluid-Solid Momentum Equations 665
14.5.4.2.4. Conservation of Energy 665
14.5.5. Surface Tension and Adhesion for the Eulerian Multiphase Model 665
14.5.6. Interfacial Area Concentration 665
14.5.7. Interphase Exchange Coefficients 667
14.5.7.1. Fluid-Fluid Exchange Coefficient 667
14.5.7.1.1. Schiller and Naumann Model 668
14.5.7.1.2. Morsi and Alexander Model 668
14.5.7.1.3. Symmetric Model 669
14.5.7.1.4. Grace et al. Model 670
14.5.7.1.5. Tomiyama et al. Model 671
14.5.7.1.6. Ishii Model 671
14.5.7.1.7. Ishii-Zuber Drag Model 672
14.5.7.1.8. Universal Drag Laws for Bubble-Liquid and Droplet-Gas Flows 673
14.5.7.1.8.1. Bubble-Liquid Flow 674
14.5.7.1.8.2. Droplet-Gas Flow 675
14.5.7.2. Fluid-Solid Exchange Coefficient 675
14.5.7.3. Solid-Solid Exchange Coefficient 680
14.5.7.4. Drag Modification 680
14.5.7.4.1. Brucato et al. Correlation 681
14.5.7.4.2. Near-Wall Drag Enhancement 681
14.5.8. Lift Coefficient Correction 681
14.5.8.1. Shaver-Podowski Correction 681
14.5.9. Lift Force 682
14.5.9.1. Lift Coefficient Models 683
14.5.9.1.1. Moraga Lift Force Model 683

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

14.5.9.1.2. Saffman-Mei Lift Force Model 683
14.5.9.1.3. Legendre-Magnaudet Lift Force Model 684
14.5.9.1.4. Tomiyama Lift Force Model 684
14.5.9.1.5. Hessenkemper et al. Lift Force Model 685
14.5.10. Wall Lubrication Force 686
14.5.10.1. Wall Lubrication Models 687
14.5.10.1.1. Antal et al. Model 687
14.5.10.1.2. Tomiyama Model 687
14.5.10.1.3. Frank Model 688
14.5.10.1.4. Hosokawa Model 688
14.5.10.1.5. Lubchenko Model 689
14.5.11. Turbulent Dispersion Force 690
14.5.11.1. Models for Turbulent Dispersion Force 690
14.5.11.1.1. Lopez de Bertodano Model 691
14.5.11.1.2. Simonin Model 691
14.5.11.1.3. Burns et al. Model 691
14.5.11.1.4. Diffusion in VOF Model 692
14.5.11.2. Limiting Functions for the Turbulent Dispersion Force 692
14.5.12. Virtual Mass Force 693
14.5.13. Solids Pressure 694
14.5.13.1. Radial Distribution Function 695
14.5.14. Maximum Packing Limit in Binary Mixtures 696
14.5.15. Solids Shear Stresses 697
14.5.15.1. Collisional Viscosity 697
14.5.15.2. Kinetic Viscosity 697
14.5.15.3. Bulk Viscosity 697
14.5.15.4. Frictional Viscosity 698
14.5.16. Granular Temperature 699
14.5.17. Description of Heat Transfer 701
14.5.17.1. The Heat Exchange Coefficient 701
14.5.17.1.1. Constant 702
14.5.17.1.2. Nusselt Number 702
14.5.17.1.3. Ranz-Marshall Model 702
14.5.17.1.4. Tomiyama Model 702
14.5.17.1.5. Hughmark Model 702
14.5.17.1.6. Gunn Model 703
14.5.17.1.7. Two-Resistance Model 703
14.5.17.1.8. Fixed To Saturation Temperature 704
14.5.17.1.9. Constant Time Scale Method 704
14.5.17.1.10. User Defined 705
14.5.18. Turbulence Models 705
14.5.18.1. k-ε Turbulence Models 705
14.5.18.1.1. k-ε Mixture Turbulence Model 705
14.5.18.1.2. k-ε Dispersed Turbulence Model 707
14.5.18.1.2.1. Assumptions 707
14.5.18.1.2.2. Turbulence in the Continuous Phase 707
14.5.18.1.2.3. Turbulence in the Dispersed Phase 708
14.5.18.1.3. k-ε Turbulence Model for Each Phase 708
14.5.18.1.3.1. Transport Equations 709
14.5.18.2. RSM Turbulence Models 709
14.5.18.2.1. RSM Dispersed Turbulence Model 710

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

14.5.18.2.2. RSM Mixture Turbulence Model 711
14.5.18.3. Turbulence Interaction Models 712
14.5.18.3.1. Simonin et al. 712
14.5.18.3.1.1. Formulation in Dispersed Turbulence Models 712
14.5.18.3.1.1.1. Continuous Phase 712
14.5.18.3.1.1.2. Dispersed Phases 713
14.5.18.3.1.2. Formulation in Per Phase Turbulence Models 714
14.5.18.3.2. Troshko-Hassan 714
14.5.18.3.2.1. Troshko-Hassan Formulation in Mixture Turbulence Models 715
14.5.18.3.2.2. Troshko-Hassan Formulation in Dispersed Turbulence Models 715
14.5.18.3.2.2.1. Continuous Phase 715
14.5.18.3.2.2.2. Dispersed Phases 715
14.5.18.3.2.3. Troshko-Hassan Formulation in Per-Phase Turbulence Models 715
14.5.18.3.2.3.1. Continuous Phase 715
14.5.18.3.2.3.2. Dispersed Phases 716
14.5.18.3.3. Sato 716
14.5.18.3.4. None 716
14.5.19. Solution Method in Ansys Fluent 716
14.5.19.1. The Pressure-Correction Equation 717
14.5.19.2. Volume Fractions 717
14.5.20. Algebraic Interfacial Area Density (AIAD) Model 717
14.5.20.1. Modeling Interfacial Area 719
14.5.20.2. Modeling Free-Surface Drag 720
14.5.20.3. Modeling Sub-grid Wave Turbulence Contribution (SWT) 721
14.5.20.4. Modeling Entrainment-Absorption 723
14.5.21. Generalized Two Phase (GENTOP) Flow Model 724
14.5.21.1. Interface Detection of the GENTOP Phase 725
14.5.21.2. Clustering Force for the GENTOP Phase 726
14.5.21.3. Surface Tension for the GENTOP-Primary Phase Pair 726
14.5.21.4. Interface Momentum Transfer 727
14.5.21.5. Complete Coalescence Method 727
14.5.22. The Filtered Two-Fluid Model 727
14.5.23. Dense Discrete Phase Model 729
14.5.23.1. Limitations 731
14.5.23.2. Granular Temperature 731
14.5.24. Multi-Fluid VOF Model 732
14.5.25. Wall Boiling Models 733
14.5.25.1. Overview of Wall Boiling Models 734
14.5.25.2. RPI Model 734
14.5.25.3. Non-equilibrium Subcooled Boiling 737
14.5.25.4. Critical Heat Flux 737
14.5.25.4.1. Wall Heat Flux Partition 738
14.5.25.4.2. Coupling Between the RPI Boiling Model and the Homogeneous or Inhomogeneous Discrete PBM 738
14.5.25.4.3. Flow Regime Transition 738
14.5.25.5. Interfacial Momentum Transfer 739
14.5.25.5.1. Interfacial Area 740
14.5.25.5.2. Bubble and Droplet Diameters 740
14.5.25.5.2.1. Bubble Diameter 740
14.5.25.5.2.2. Droplet Diameter 740
14.5.25.5.3. Interfacial Drag Force 741

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

14.5.25.5.4. Interfacial Lift Force 741
14.5.25.5.5. Turbulent Dispersion Force 741
14.5.25.5.6. Wall Lubrication Force 741
14.5.25.5.7. Virtual Mass Force 741
14.5.25.6. Mass Transfer 742
14.5.25.6.1. Mass Transfer From the Wall to Vapor 742
14.5.25.6.2. Interfacial Mass Transfer 742
14.5.25.7. Turbulence Interactions 742
14.6. Wet Steam Model Theory 742
14.6.1. Overview of the Wet Steam Model 742
14.6.2. Limitations of the Wet Steam Model 743
14.6.3. Wet Steam Flow Equations 743
14.6.4. Phase Change Model 744
14.6.5. Built-in Thermodynamic Wet Steam Properties 746
14.6.5.1. Equation of State 747
14.6.5.1.1. Virial Equation Developed by Vukalovich 747
14.6.5.1.2. Virial Equation Developed by Young 748
14.6.5.2. Saturated Vapor Line 749
14.6.5.3. Saturated Liquid Line 749
14.6.5.4. Mixture Properties 749
14.6.6. Real Gas Property (RGP) Table Files for the Wet Steam Model 749
14.6.7. Computing Stagnation Conditions for the Wet Steam Model 750
14.7. Modeling Mass Transfer in Multiphase Flows 751
14.7.1. Source Terms due to Mass Transfer 751
14.7.1.1. Mass Equation 751
14.7.1.2. Momentum Equation 751
14.7.1.3. Energy Equation 752
14.7.1.4. Species Equation 752
14.7.1.5. Other Scalar Equations 752
14.7.2. Unidirectional Constant Rate Mass Transfer 752
14.7.3. UDF-Prescribed Mass Transfer 753
14.7.4. Cavitation Models 753
14.7.4.1. Limitations of the Cavitation Models 753
14.7.4.2. Vapor Transport Equation 754
14.7.4.3. Bubble Dynamics Consideration 754
14.7.4.4. Schnerr and Sauer Model 755
14.7.4.5. Zwart-Gerber-Belamri Model 757
14.7.4.6. Expert Cavitation Model 758
14.7.4.6.1. Limitations of the Singhal et al. Model 758
14.7.4.6.2. Singhal et al. Model Theory 758
14.7.4.7. Turbulence Factor 760
14.7.4.8. Additional Guidelines for the Cavitation Models 760
14.7.4.9. Extended Cavitation Model Capabilities 762
14.7.4.9.1. N-phase Cavitation Models 762
14.7.4.9.2. Multiphase Species Transport Cavitation Model 763
14.7.5. Evaporation-Condensation Model 763
14.7.5.1. Lee Model 763
14.7.5.2. Thermal Phase Change Model 766
14.7.6. Semi-Mechanistic Boiling Model 767
14.7.7. Interphase Species Mass Transfer 770
14.7.7.1. Modeling Approach 771

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

14.7.7.2. Equilibrium Models 773
14.7.7.2.1. Raoult's Law 774
14.7.7.2.2. Henry's Law 775
14.7.7.2.3. Equilibrium Ratio 776
14.7.7.3. Mass Transfer Coefficient Models 776
14.7.7.3.1. Constant 777
14.7.7.3.2. Sherwood Number 777
14.7.7.3.3. Ranz-Marshall Model 777
14.7.7.3.4. Hughmark Model 777
14.7.7.3.5. Higbie Model 778
14.7.7.3.6. User-Defined 778
14.8. Modeling Species Transport in Multiphase Flows 778
14.8.1. Limitations 779
14.8.2. Mass and Momentum Transfer with Multiphase Species Transport 779
14.8.2.1. Source Terms Due to Heterogeneous Reactions 780
14.8.2.1.1. Mass Transfer 780
14.8.2.1.2. Momentum Transfer 780
14.8.2.1.3. Species Transfer 781
14.8.2.1.4. Heat Transfer 781
14.8.3. The Stiff Chemistry Solver 782
14.8.4. Heterogeneous Phase Interaction 782
15. Population Balance Model 785
15.1. Introduction 785
15.1.1. The Discrete Method 785
15.1.2. The Inhomogeneous Discrete Method 785
15.1.3. The Standard Method of Moments 787
15.1.4. The Quadrature Method of Moments 788
15.2. Population Balance Model Theory 788
15.2.1. The Particle State Vector 788
15.2.2. The Population Balance Equation (PBE) 789
15.2.2.1. Particle Growth 789
15.2.2.2. Particle Birth and Death Due to Breakage and Aggregation 790
15.2.2.2.1. Breakage 790
15.2.2.2.2. Luo and Lehr Breakage Kernels 792
15.2.2.2.3. Ghadiri Breakage Kernels 792
15.2.2.2.4. Laakkonen Breakage Kernels 793
15.2.2.2.5. Liao Breakage Kernel 793
15.2.2.2.6. Parabolic PDF 795
15.2.2.2.7. Generalized PDF 795
15.2.2.2.8. Martinez-Bazan Breakage Kernel 798
15.2.2.2.9. Inverted U-PDF 799
15.2.2.2.10. Aggregation 799
15.2.2.2.11. Luo Aggregation Kernel 800
15.2.2.2.12. Free Molecular Aggregation Kernel 801
15.2.2.2.13. Turbulent Aggregation Kernel 801
15.2.2.2.14. Prince and Blanch Aggregation Kernel 802
15.2.2.2.15. Liao Aggregation Kernel 803
15.2.2.3. Particle Birth by Nucleation 806
15.2.3. Solution Methods 807
15.2.3.1. The Discrete Method and the Inhomogeneous Discrete Method 807
15.2.3.1.1. Numerical Method 807

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

15.2.3.1.2. Breakage Formulations for the Discrete Method 809
15.2.3.2. The Standard Method of Moments (SMM) 810
15.2.3.2.1. Numerical Method 810
15.2.3.3. The Quadrature Method of Moments (QMOM) 811
15.2.3.3.1. Numerical Method 811
15.2.3.4. The Direct Quadrature Method of Moments (DQMOM) 812
15.2.3.4.1. Numerical Method 813
15.2.4. Population Balance Statistics 814
15.2.4.1. Reconstructing the Particle Size Distribution from Moments 814
15.2.4.2. The Log-Normal Distribution 815
16. Solidification and Melting 817
16.1. Overview 817
16.2. Limitations 818
16.3. Introduction 818
16.4. Energy Equation 819
16.5. Momentum Equations 820
16.6. Turbulence Equations 820
16.7. Species Equations 820
16.8. Back Diffusion 822
16.9. Pull Velocity for Continuous Casting 823
16.10. Contact Resistance at Walls 824
16.11. Thermal and Solutal Buoyancy 824
17. The Structural Model for Intrinsic Fluid-Structure Interaction (FSI) 827
17.1. Limitations 827
17.2. The FSI Model 828
17.3. Intrinsic FSI 828
17.4. Linear Elasticity 829
17.4.1. Equations 829
17.4.1.1. Linear Isotropic and Isothermal Elasticity 829
17.4.1.2. Evaluation of the von Mises Stress 829
17.4.2. Finite Element Representation 830
17.4.2.1. Construction of the Matrix of the System 830
17.4.2.2. Dynamic Structural Systems 831
17.4.2.2.1. The Newmark Method 831
17.4.2.2.2. Backward Euler Method 832
17.4.2.2.3. Rayleigh Damping 833
17.5. Nonlinear Elasticity 833
17.5.1. Finite Element Geometric Nonlinearity 833
17.5.2. Finite Element Nonlinear Discretization 834
17.5.3. Constitutive Equations 835
17.5.4. The Transient Scheme for the Nonlinear System 835
17.6. Thermoelasticity Model 836
17.6.1. Constitutive Equations 836
17.6.2. Finite Element Discretization 836
17.7. Coupling of the Structural Model with the Battery Swelling Model 837
18. Eulerian Wall Films 839
18.1. Introduction 839
18.2. Mass, Momentum, and Energy Conservation Equations for Wall Film 840
18.2.1. Film Sub-Models 841
18.2.1.1. DPM Collection 841
18.2.1.2. Particle-Wall Interaction 842

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

18.2.1.3. Film Separation 842
18.2.1.3.1. Separation Criteria 842
18.2.1.3.1.1. Foucart Separation 842
18.2.1.3.1.2. O'Rourke Separation 843
18.2.1.3.1.3. Friedrich Separation 843
18.2.1.4. Film Stripping 844
18.2.1.5. Secondary Phase Accretion 845
18.2.1.6. Coupling of Wall Film with Mixture Species Transport 845
18.2.1.7. Coupling of Eulerian Wall Film with the VOF Multiphase Model 846
18.2.2. Partial Wetting Effect 847
18.2.3. Boundary Conditions 847
18.2.4. Obtaining Film Velocity Without Solving the Momentum Equations 847
18.2.4.1. Shear-Driven Film Velocity 848
18.2.4.2. Gravity-Driven Film Velocity 848
18.3. Passive Scalar Equation for Wall Film 849
18.4. Numerical Schemes and Solution Algorithm 850
18.4.1. Temporal Differencing Schemes 850
18.4.1.1. First-Order Explicit Method 850
18.4.1.2. First-Order Implicit Method 851
18.4.1.3. Second-Order Implicit Method 852
18.4.2. Spatial Differencing Schemes 852
18.4.3. Solution Algorithm 853
18.4.3.1. Steady Flow 853
18.4.3.2. Transient Flow 854
18.4.4. Coupled Solution Approach 854
19. Electric Potential and Electrochemistry Models 857
19.1. Electric Potential 857
19.1.1. Overview 857
19.1.2. Electric Potential Equation 858
19.1.3. Energy Equation Source Term 858
19.2. Lithium-ion Battery Model 858
19.2.1. Overview 858
19.2.2. Lithium-ion Battery Model Theory 859
19.3. Electrolysis and H2 Pump Model 862
19.3.1. Overview 862
19.3.2. Resolved Modeling Approach 863
19.3.2.1. Electrochemistry Modeling 864
19.3.2.2. Multiphase Modeling 864
19.3.2.3. Heat Source 867
19.3.3. Unresolved 0D Modeling Approach 868
20. Battery Model 871
20.1. Battery Solution Methods 871
20.1.1. CHT Coupling Method 872
20.1.2. FMU-CHT coupling method 872
20.1.3. Circuit Network Solution Method 872
20.1.4. MSMD Solution Method 873
20.2. Electro-Chemical Models 874
20.2.1. NTGK/DCIR Model 874
20.2.2. ECM Model 875
20.2.3. Newman's P2D Model 877
20.3. MSMD Solution Method: Coupling Between CFD and Submodels 881

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

20.4. Simulating Battery Pack Using the MSMD Solution Method 881
20.5. Reduced Order Solution Method (ROM) 883
20.6. External and Internal Electric Short-Circuit Treatment 884
20.7. Thermal Abuse Model 885
20.8. Battery Life Models 887
20.8.1. Empirical Battery Life Model 887
20.8.2. Battery Capacity Fade Model 888
20.8.3. Physics-Based Battery Life Model 888
20.9. Battery Swelling Model 892
20.9.1. Empirical-Based Swelling Model 892
20.9.2. Physics-Based Swelling Model 892
20.9.3. Coupling between the Swelling and FSI Structural Models 895
20.9.4. Modeling Swelling in the E-Chem Standalone Mode 897
21. Modeling Fuel Cells 899
21.1. PEMFC Model Theory 899
21.1.1. Introduction 899
21.1.2. Electrochemistry Modeling 901
21.1.2.1. The Cathode Particle Model 904
21.1.3. Current and Mass Conservation 905
21.1.4. Water Transport and Mass Transfer in PEMFC 906
21.1.4.1. The Dissolved Phase Model 907
21.1.4.2. The Liquid Phase Model 908
21.1.4.2.1. Liquid Water Transport Equation in the Porous Electrode and the Membrane 908
21.1.4.2.2. Liquid Water Transport Equation in Gas Channels 909
21.1.5. Heat Source 910
21.1.6. Properties 910
21.1.7. Transient Simulations 913
21.1.8. Leakage Current (Cross-Over Current) 913
21.1.9. Zones Where User-Defined Scalars Have Physically Meaningful Values 914
21.2. Fuel Cell and Electrolysis Model Theory 914
21.2.1. Introduction 914
21.2.1.1. Introduction to PEMFC 916
21.2.1.1.1. Low-Temperature PEMFC 916
21.2.1.1.2. High-Temperature PEMFC 916
21.2.1.2. Introduction to SOFC 916
21.2.1.3. Introduction to Electrolysis 916
21.2.2. Electrochemistry Modeling 917
21.2.3. Current and Mass Conservation 920
21.2.4. Heat Source 920
21.2.5. Liquid Water Formation, Transport, and its Effects (Low-Temperature PEMFC Only) 921
21.2.6. Properties 922
21.2.7. Transient Simulations 924
21.2.8. Leakage Current (Cross-Over Current) 924
21.3. SOFC Fuel Cell With Unresolved Electrolyte Model Theory 925
21.3.1. Introduction 925
21.3.2. The SOFC With Unresolved Electrolyte Modeling Strategy 926
21.3.3. Modeling Fluid Flow, Heat Transfer, and Mass Transfer 927
21.3.4. Modeling Current Transport and the Potential Field 928
21.3.4.1. Cell Potential 929
21.3.4.2. Activation Overpotential 930
21.3.4.3. Treatment of the Energy Equation at the Electrolyte Interface 930

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

21.3.4.4. Treatment of the Energy Equation in the Conducting Regions 932
21.3.5. Modeling Reactions 932
21.3.5.1. Modeling Electrochemical Reactions 932
21.3.5.2. Modeling CO Electrochemistry 933
21.3.5.3. Modeling Electrolysis 933
22. Modeling Magnetohydrodynamics 935
22.1. Introduction 935
22.2. Magnetic Induction Method 936
22.2.1. Case 1: Externally Imposed Magnetic Field Generated in Non-conducting Media 936
22.2.2. Case 2: Externally Imposed Magnetic Field Generated in Conducting Media 937
22.3. Electric Potential Method 937
23. Modeling Continuous Fibers 939
23.1. Introduction 939
23.2. Governing Equations of Fiber Flow 939
23.3. Discretization of the Fiber Equations 942
23.3.1. Under-Relaxation 943
23.4. Numerical Solution Algorithm of Fiber Equations 943
23.5. Residuals of Fiber Equations 943
23.6. Coupling Between Fibers and the Surrounding Fluid 944
23.6.1. Momentum Exchange 945
23.6.2. Mass Exchange 946
23.6.3. Heat Exchange 946
23.6.4. Radiation Exchange 946
23.6.5. Under-Relaxation of the Fiber Exchange Terms 947
23.7. Fiber Grid Generation 947
23.8. Correlations for Momentum, Heat and Mass Transfer 948
23.8.1. Drag Coefficient 948
23.8.2. Heat Transfer Coefficient 950
23.8.3. Mass Transfer Coefficient 951
23.9. Fiber Properties 952
23.9.1. Fiber Viscosity 952
23.9.1.1. Melt Spinning 952
23.9.1.2. Dry Spinning 953
23.9.2. Vapor-Liquid Equilibrium 953
23.9.3. Latent Heat of Vaporization 953
23.9.4. Emissivity 953
23.10. Solution Strategies 953
24. Solver Theory 955
24.1. Overview of Flow Solvers 955
24.1.1. Pressure-Based Solver 956
24.1.1.1. The Pressure-Based Segregated Algorithm 956
24.1.1.2. The Pressure-Based Coupled Algorithm 957
24.1.2. Density-Based Solver 958
24.2. General Scalar Transport Equation: Discretization and Solution 960
24.2.1. Solving the Linear System 962
24.3. Discretization 962
24.3.1. Spatial Discretization 963
24.3.1.1. First-Order Upwind Scheme 963
24.3.1.2. Second-Order Upwind Scheme 963
24.3.1.3. First- to Higher-Order Blending 964
24.3.1.4. Central-Differencing Scheme 964

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

24.3.1.5. Bounded Central Differencing Scheme 965
24.3.1.6. QUICK Scheme 966
24.3.1.7. Third-Order MUSCL Scheme 966
24.3.1.8. Modified HRIC Scheme 967
24.3.1.9. High Order Term Relaxation 968
24.3.2. Temporal Discretization 968
24.3.2.1. Implicit Time Integration 969
24.3.2.2. Bounded Second-Order Implicit Time Integration 969
24.3.2.2.1. Limitations 970
24.3.2.3. Second-Order Time Integration Using a Variable Time Step Size 970
24.3.2.4. Explicit Time Integration 971
24.3.3. Evaluation of Gradients and Derivatives 972
24.3.3.1. Green-Gauss Theorem 972
24.3.3.2. Green-Gauss Cell-Based Gradient Evaluation 972
24.3.3.3. Green-Gauss Node-Based Gradient Evaluation 973
24.3.3.4. Least Squares Cell-Based Gradient Evaluation 973
24.3.4. Gradient Limiters 974
24.3.4.1. Standard Limiter 975
24.3.4.2. Multidimensional Limiter 975
24.3.4.3. Differentiable Limiter 976
24.4. Pressure-Based Solver 976
24.4.1. Discretization of the Momentum Equation 976
24.4.1.1. Pressure Interpolation Schemes 977
24.4.2. Discretization of the Continuity Equation 979
24.4.2.1. Density Interpolation Schemes 979
24.4.3. Pressure-Velocity Coupling 980
24.4.3.1. Segregated Algorithms 980
24.4.3.1.1. SIMPLE 980
24.4.3.1.2. SIMPLEC 981
24.4.3.1.2.1. Skewness Correction 982
24.4.3.1.3. PISO 982
24.4.3.1.3.1. Neighbor Correction 982
24.4.3.1.3.2. Skewness Correction 982
24.4.3.1.3.3. Skewness - Neighbor Coupling 983
24.4.3.1.4. Fractional-Step Method (FSM) 983
24.4.3.2. Coupled Algorithm 983
24.4.3.2.1. Limitation 984
24.4.4. Steady-State Iterative Algorithm 984
24.4.4.1. Under-Relaxation of Variables 985
24.4.4.2. Under-Relaxation of Equations 985
24.4.5. Time-Advancement Algorithm 985
24.4.5.1. Iterative Time-Advancement Scheme 986
24.4.5.1.1. The Frozen Flux Formulation 987
24.4.5.2. Non-Iterative Time-Advancement Scheme 988
24.4.6. Correction Form Discretization of the Momentum Equations 990
24.5. Density-Based Solver 990
24.5.1. Governing Equations in Vector Form 991
24.5.2. Preconditioning 991
24.5.3. Convective Fluxes 994
24.5.3.1. Roe Flux-Difference Splitting Scheme 994
24.5.3.2. AUSM+ Scheme 994

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

24.5.3.3. Low Diffusion Roe Flux Difference Splitting Scheme 995
24.5.4. Steady-State Flow Solution Methods 995
24.5.4.1. Explicit Formulation 996
24.5.4.1.1. Implicit Residual Smoothing 996
24.5.4.2. Implicit Formulation 997
24.5.4.2.1. Convergence Acceleration for Stretched Meshes 997
24.5.5. Unsteady Flows Solution Methods 998
24.5.5.1. Explicit Time Stepping 998
24.5.5.2. Implicit Time Stepping (Dual-Time Formulation) 998
24.6. Pseudo Time Method Under-Relaxation 1000
24.6.1. Local Time Step Method 1001
24.6.2. Global Time Step Method 1001
24.7. Multigrid Method 1003
24.7.1. Approach 1004
24.7.1.1. The Need for Multigrid 1004
24.7.1.2. The Basic Concept in Multigrid 1004
24.7.1.3. Restriction and Prolongation 1005
24.7.1.4. Unstructured Multigrid 1005
24.7.2. Multigrid Cycles 1005
24.7.2.1. The V and W Cycles 1006
24.7.3. Algebraic Multigrid (AMG) 1009
24.7.3.1. AMG Restriction and Prolongation Operators 1009
24.7.3.2. AMG Coarse Level Operator 1010
24.7.3.3. The F Cycle 1010
24.7.3.4. The Flexible Cycle 1011
24.7.3.4.1. The Residual Reduction Rate Criteria 1011
24.7.3.4.2. The Termination Criteria 1012
24.7.3.5. The Coupled and Scalar AMG Solvers 1012
24.7.3.5.1. Gauss-Seidel 1013
24.7.3.5.2. Incomplete Lower Upper (ILU) 1013
24.7.4. Full-Approximation Storage (FAS) Multigrid 1014
24.7.4.1. FAS Restriction and Prolongation Operators 1015
24.7.4.2. FAS Coarse Level Operator 1015
24.8. Hybrid Initialization 1016
24.9. Full Multigrid (FMG) Initialization 1018
24.9.1. Overview of FMG Initialization 1018
24.9.2. Limitations of FMG Initialization 1019
25. Adapting the Mesh 1021
25.1. Adaption Process 1021
25.1.1. Hanging Node Adaption 1022
25.1.2. Polyhedral Unstructured Mesh Adaption 1024
25.2. Geometry-Based Adaption 1025
25.2.1. Geometry-Based Adaption Approach 1025
25.2.1.1. Node Projection 1025
25.2.1.2. Example of Geometry-Based Adaption 1028
26. Reporting Alphanumeric Data 1031
26.1. Fluxes Through Boundaries 1031
26.2. Forces on Boundaries 1032
26.2.1. Computing Forces, Moments, and the Center of Pressure 1033
26.3. Surface Integration 1036
26.3.1. Computing Surface Integrals 1037

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

26.3.1.1. Area 1037
26.3.1.2. Integral 1037
26.3.1.3. Area-Weighted Average 1037
26.3.1.4. Custom Vector Based Flux 1037
26.3.1.5. Custom Vector Flux 1038
26.3.1.6. Custom Vector Weighted Average 1038
26.3.1.7. Flow Rate 1038
26.3.1.8. Mass Flow Rate 1038
26.3.1.9. Mass-Weighted Average 1039
26.3.1.10. Sum of Field Variable 1040
26.3.1.11. Facet Average 1040
26.3.1.12. Facet Minimum 1040
26.3.1.13. Facet Maximum 1040
26.3.1.14. Vertex Average 1040
26.3.1.15. Vertex Minimum 1040
26.3.1.16. Vertex Maximum 1041
26.3.1.17. Standard-Deviation 1041
26.3.1.18. Uniformity Index 1041
26.3.1.19. Volume Flow Rate 1042
26.4. Volume Integration 1042
26.4.1. Computing Volume Integrals 1043
26.4.1.1. Volume 1043
26.4.1.2. Sum 1043
26.4.1.3. Sum*2Pi 1043
26.4.1.4. Volume Integral 1043
26.4.1.5. Volume-Weighted Average 1044
26.4.1.6. Mass-Weighted Integral 1044
26.4.1.7. Mass 1044
26.4.1.8. Mass-Weighted Average 1044
26.5. Efficiency Calculation 1045
26.5.1. Isentropic Efficiency 1045
26.5.2. Polytropic Efficiency 1046
A. Nomenclature 1049
Bibliography 1053

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information
of ANSYS, Inc. and its subsidiaries and affiliates.

List of Figures

1.1. Example of Periodic Flow in a 2D Heat Exchanger Geometry 7
1.2. Example of a Periodic Geometry 8
1.3. Rotating Flow in a Cavity 10
1.4. Swirling Flow in a Gas Burner 10
1.5. Typical Radial Distribution of Circumferential Velocity in a Free Vortex 11
1.6. Stream Function Contours for Rotating Flow in a Cavity 12
1.7. Transonic Flow in a Converging-Diverging Nozzle 13
1.8. Mach 0.675 Flow Over a Bump in a 2D Channel 13
2.1. Single Component (Blower Wheel Blade Passage) 20
2.2. Multiple Component (Blower Wheel and Casing) 20
2.3. Stationary and Moving Reference Frames 21
2.4. Geometry with One Rotating Impeller 25
2.5. Geometry with Two Rotating Impellers 26
2.6. Interface Treatment for the MRF Model 27
2.7. Axial Rotor-Stator Interaction (Schematic Illustrating the Mixing Plane Concept) 29
2.8. Radial Rotor-Stator Interaction (Schematic Illustrating the Mixing Plane Concept) 29
3.1. A Mesh Associated With Moving Pistons 36
3.2. Blower 37
4.1. Effect of Increasing y+ for the Flat Plate T3A Test Case 81
4.2. Effect of Decreasing y+ for the Flat Plate T3A Test Case 82
4.3. Effect of Wall Normal Expansion Ratio for the Flat Plate T3A Test Case 83
4.4. Effect of Streamwise Mesh Density for the Flat Plate T3A Test Case 83
4.5. Exemplary Decay of Turbulence Intensity (Tu) as a Function of Streamwise Distance (x) 85
4.6. Resolved Structures for Cylinder in Cross Flow (top: URANS; bottom: SST-SAS) 107
4.7. Eddy Viscosity Profiles 116
4.8. The Computational Domain and Mesh for the Subsonic Jet Flow 118
4.9. Iso-Surfaces of the Q-Criterion Colored with the Velocity Magnitude 119
4.10. Distribution of the Mean (Left) and RMS (Right) Velocity along the Jet Centerline 119
4.11. Turbulent Structures Generated by Synthetic Turbulence Generator in the Boundary Layer with Arbitrarily High Turbulent Length Scale in the Free Stream: With Scale Search Limiter Option Set to None (left) and with Specified Turbulent Intensity Threshold 2% (right) 130
4.12. Splitting of a cell zone in a complex flow simulation to avoid the undesirable side effect of the volumetric forcing 132
4.13. Backward Facing Step Flow Using ELES 133
4.14. Typical Grid for ELES for Backward Facing Step 135
4.15. Subdivisions of the Near-Wall Region 136
4.16. Near-Wall Treatments in Ansys Fluent 137
4.17. Computational meshes for low and high Reynolds number cases 152
4.18. Comparison of the skin friction distribution on a fine (y+ ~ 0.1, left) and coarse (y+ ~ 22, right) mesh for the 'correlation' and 'tabulated' near wall treatments with SST. 155
4.19. Downward Shift of the Logarithmic Velocity Profile 160
4.20. Illustration of Equivalent Sand-Grain Roughness 161
5.1. Radiative Heat Transfer 189
5.2. Angles θ and ϕ Defining the Hemispherical Solid Angle About a Point P 196
5.3. Angular Coordinate System 201
5.4. Face with No Control Angle Overhang 201
5.5. Face with Control Angle Overhang 202
5.6. Face with Control Angle Overhang (3D) 202
5.7. Pixelation of Control Angle 203

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

5.8. DO Radiation on Opaque Wall 205
5.9. DO Radiation on Interior Semi-Transparent Wall 208
5.10. Reflection and Refraction of Radiation at the Interface Between Two Semi-Transparent Media 209
5.11. Critical Angle θc 210
5.12. DO Irradiation on External Semi-Transparent Wall 212
5.13. Beam Width and Direction for External Irradiation Beam 213
6.1. An Example of a Four-Pass Heat Exchanger 224
6.2. Core Discretized into 3x4x2 Macros 225
6.3. Core with Matching Quad Meshes for Primary and Auxiliary Zones in a Crossflow Pattern 233
6.4. Core with Primary and Auxiliary Zones with Overlap of Cells 234
7.1. A Reacting Particle in the Multiple Surface Reactions Model 262
7.2. Cross-section of a Channel and Outer Shell Around It. 271
8.1. Relationship of Mixture Fractions (Fuel, Secondary Stream, and Oxidizer) 297
8.2. Relationship of Mixture Fractions (Fuel, Secondary Stream, and Normalized Secondary Mixture Fraction) 297
8.3. Graphical Description of the Probability Density Function 302
8.4. Example of the Double Delta Function PDF Shape 304
8.5. Logical Dependence of Averaged Scalars on Mean Mixture Fraction, the Mixture Fraction Variance, and the Chemistry Model (Adiabatic, Single-Mixture-Fraction Systems) 305
8.6. Logical Dependence of Averaged Scalars on Mean Mixture Fraction, the Mixture Fraction Variance, Mean Enthalpy, and the Chemistry Model (Non-Adiabatic, Single-Mixture-Fraction Systems) 306
8.7. Reacting Systems Requiring Non-Adiabatic Non-Premixed Model Approach 307
8.8. Visual Representation of a Look-Up Table for the Scalar (Mean Value of Mass Fractions, Density, or Temperature) as a Function of Mean Mixture Fraction and Mixture Fraction Variance in Adiabatic Single-Mixture-Fraction Systems 308
8.9. Visual Representation of a Look-Up Table for the Scalar ϕ_I as a Function of Fuel Mixture Fraction and Secondary Partial Fraction in Adiabatic Two-Mixture-Fraction Systems 309
8.10. Visual Representation of a Look-Up Table for the Scalar as a Function of Mean Mixture Fraction and Mixture Fraction Variance and Normalized Heat Loss/Gain in Non-Adiabatic Single-Mixture-Fraction Systems 310
8.11. Visual Representation of a Look-Up Table for the Scalar ϕ_I as a Function of Fuel Mixture Fraction and Secondary Partial Fraction, and Normalized Heat Loss/Gain in Non-Adiabatic Two-Mixture-Fraction Systems 311
8.12. Chemical Systems That Can Be Modeled Using a Single Mixture Fraction 315
8.13. Chemical System Configurations That Can Be Modeled Using Two Mixture Fractions 316
8.14. Premixed Systems That Cannot Be Modeled Using the Non-Premixed Model 316
8.15. Using the Non-Premixed Model with Flue Gas Recycle 317
8.16. Laminar Opposed-Flow Diffusion Flamelet 320
8.17. The Scalar Dissipation Rate Along The Normalized Reaction Progress Variable 349
8.18. Premix Opposed Flow Configuration for the Strained Flame Speed 358
9.1. De Soete's Global NOx Mechanism with Additional Reduction Path 384
9.2. Simplified Reaction Mechanism for the SNCR Process 387
10.1. Flame Front Showing Accumulation of Source Terms for the Knock Model 425
10.2. Propagating Fuel Cloud Showing Accumulation of Source Terms for the Ignition Delay Model 426
10.3. Crevice Model Geometry (Piston) 428
10.4. Crevice Model Geometry (Ring) 428
10.5. Crevice Model "Network" Representation 428
11.1. Schematic of the Convective Effect on the Retarded Time Calculation 436
12.1. Coal Bridge 476
12.2. Particle Reflection at Wall 492
12.3. Particle-Wall Collision Forces 493

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

12.4. Particle-Wall Interaction at a Rough Wall 495
12.5. Wall Roughness Parameters 495
12.6. "Wall Jet" Boundary Condition for the Discrete Phase 497
12.7. Mechanisms of Splashing, Momentum, Heat and Mass Transfer for the Lagrangian Wall-Film 498
12.8. Simplified Decision Chart for Wall Interaction Criterion 502
12.9. The Stanton-Rutland Model: Impinging and Splashing 504
12.10. The Kuhnke Impingement Model Regimes 508
12.11. The Kuhnke Model: Impinging Drop 513
12.12. The Regime Map of the Stochastic Kuhnke Model 515
12.13. Assumption of a Bilinear Temperature Profile in the Film 525
12.14. Convective Wall-Film Heat Transfer: Film Flow Over a Heated Surface 527
12.15. Fully Turbulent Film Flow Regime: Velocity and Temperature Profile Assumptions 531
12.16. Geometric Parameters of Deformed Impinging Droplet in Heat Transfer Calculations 540
12.17. Single-Phase Nozzle Flow (Liquid Completely Fills the Orifice) 542
12.18. Cavitating Nozzle Flow (Vapor Pockets Form Just After the Inlet Corners) 543
12.19. Flipped Nozzle Flow (Downstream Gas Surrounds the Liquid Jet Inside the Nozzle) 543
12.20. Decision Tree for the State of the Cavitating Nozzle 545
12.21. Theoretical Progression from the Internal Atomizer Flow to the External Spray 548
12.22. Flat Fan Viewed from Above and from the Side 553
12.23. Liquid Core Approximation 563
12.24. Madabhushi Breakup Model 566
12.25. Child Droplet Velocity 568
12.26. Madabhushi Diameter Distribution 569
12.27. Particles Represented by Spheres 577
12.28. An Example of a Friction Coefficient Plot 581
12.29. Force Evaluation for Parcels 582
12.30. Heat, Mass, and Momentum Transfer Between the Discrete and Continuous Phases 583
13.1. Fixing Velocities in Fluid Cells Touched by the Particle 590
14.1. Multiphase Flow Regimes 597
14.2. Interface Calculations 608
14.3. Free Surface Positions With and Without Wall Adhesion 616
14.4. Typical Wave Spectrum 632
14.5. Schematic View of the Interface Cut Through the Front Cell 639
14.6. Distance to the Interface Segment 640
14.7. Hybrid Phase Morphology with no Regime Transition 654
14.8. Hybrid Phase Morphology with Regime Transition 654
14.9. Weighting Functions f_i and f_j 658
14.10. Weighting Function f_s 659
14.11. Length-Turbulence Velocity Diagram for Water 722
14.12. Droplet Entrainment at the Interface 723
14.13. The Stability Phase Diagram 764
14.14. The Boiling Curve 767
14.15. Distribution of Molar Concentration in the Two-Resistance Model 772
15.1. Homogeneous Discrete Method 787
15.2. Inhomogeneous Discrete Method 787
15.3. Bubble Breakup 793
15.4. Bubble Breakup Mechanisms in a Turbulent Flow 794
15.5. Bubble Collision in a Turbulent Flow 804
15.6. A Particle Size Distribution as Represented by the Discrete Method 807
15.7. Reconstruction of a Particle Size Distribution 815

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

16.1. "Pulling" a Solid in Continuous Casting 823
16.2. Circuit for Contact Resistance 824
18.1. Subgrid Processes That Require a Wall Film Model 840
18.2. Separation Criteria 842
18.3. Shear-Driven Film Velocity 848
18.4. Gravity-Driven Film Velocity 849
18.5. Spatial Gradient 853
18.6. Face Value Smoothing 855
19.1. Schematic structure of an electrode pair 859
19.2. Schematic of an Electrolysis Device 862
20.1. Electric Circuits Used in the ECM Model 876
20.2. Electrode and Particle Domains in the Newman's Model 878
20.3. Solution Domain for Two Potential Equations in a Battery Pack System 882
20.4. Electrochemical Reactions on the Particle Surface 889
20.5. Physics-Based Battery Swelling Model: Reference and Deformed Configurations 893
20.6. Coupling of the Swelling and FSI Structural Models 896
21.1. Schematic of a PEM Fuel Cell 900
21.2. Boundary Conditions for the Electric Potentials (Solid and Membrane) — PEM Fuel Cell 902
21.3. Schematic of a PEM Fuel Cell 915
21.4. Boundary Conditions for the Electric Potential (Solid and Membrane) — PEM Fuel Cell 918
21.5. Schematic of a Solid Oxide Fuel Cell 925
21.6. How the SOFC With Unresolved Electrolyte Model Works in Ansys Fluent 927
21.7. Energy Balance at the Electrolyte Interface 931
23.1. Fiber Grid Penetrating Grid of the Gas Flow 945
23.2. Dimensionless Groups of Drag Coefficient and Nusselt Number 949
24.1. Overview of the Pressure-Based Solution Methods 957
24.2. Overview of the Density-Based Solution Method 959
24.3. Control Volume Used to Illustrate Discretization of a Scalar Transport Equation 962
24.4. One-Dimensional Control Volume 966
24.5. Cell Representation for Modified HRIC Scheme 967
24.6. Cell Centroid Evaluation 973
24.7. Overview of the Iterative Time Advancement Solution Method For the Segregate Solver 987
24.8. Overview of the Non-Iterative Time Advancement Solution Method 989
24.9. V-Cycle Multigrid 1006
24.10. W-Cycle Multigrid 1007
24.11. Logic Controlling the Flex Multigrid Cycle 1011
24.12. Node Agglomeration to Form Coarse Grid Cells 1015
24.13. The FMG Initialization 1019
25.1. Example of a Hanging Node 1022
25.2. Hanging Node Adaption of 2D Cell Types 1023
25.3. Hanging Node Adaption of 3D Cell Types 1023
25.4. PUMA Refinement of a Polyhedral Cell 1024
25.5. Mesh Before Adaption 1026
25.6. Projection of Nodes 1026
25.7. Levels Projection Propagation and Magnitude 1027
25.8. Coarse Mesh of a Sphere 1028
25.9. Adapted Mesh Without Geometry Reconstruction 1029
25.10. Mesh after Geometry-Based Adaption 1030
26.1. Moment About a Specified Moment Center 1034
26.2. T-s Diagrams for Compression and Expansion Processes 1045

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

List of Tables

1. Mini Flow Chart Symbol Descriptions xliii
4.1. Wall-Resolved Grid Size as a Function of Reynolds Number 125
4.2. WMLES Grid Size as a Function of Reynolds Number 125
4.3. Error in drag prediction on a fine (y+ ~ 0.1) and coarse mesh (y+ ~ 22) for the 'correlation' and 'tabulated' near wall treatments with SST, relative to (y+ ~ 0.001) mesh 155
9.1. Rate Constants for Different Reburn Fuels 386
9.2. Seven-Step Reduced Mechanism for SNCR with Urea 388
9.3. Two-Step Urea Breakdown Process 389
9.4. Sticking Coefficient for Different PAH Species 405
9.5. Arrhenius rate parameters for HACA mechanism 410
10.1. Default Values of the Variables in the Hardenburg Correlation 427
12.1. Chemical Structure Parameters for C NMR for 13 Coals 479
12.2. Parameter *A* as a Function of Wall Roughness 516
12.3. Example of the Oka Erosion Model Constants 535
12.4. Example of the McLaury Erosion Model Constants 536
12.5. Recommended Values for Material Constants 536
12.6. Model Constants 537
12.7. List of Governing Parameters for Internal Nozzle Flow 543
12.8. Values of Spread Parameter for Different Nozzle States 547
12.9. Comparison of a Spring-Mass System to a Distorting Droplet 556
14.1. Slope Limiter Values and Their Discretization Schemes 610
14.2. Flow Regime Definitions 654
14.3. Fitting Formulas for Index H_D 679
15.1. Luo Model Parameters 792
15.2. Lehr Model Parameters 792
15.3. Daughter Distributions 797
15.4. Daughter Distributions (cont.) 797
15.5. Values for Daughter Distributions in General Form 798
19.1. Source Terms for Species and Liquid Water due to Electrochemistry Reactions 866
19.2. Volumetric Heat Source Terms 867
19.3. Source Terms for Species and Liquid Water due to Electrochemistry Reactions in the Unresolved 0D Approach 869
21.1. Volumetric Heat Source Terms 910
21.2. Zones where UDSs are Solved in PEMFC 914
24.1. Summary of the Density-Based Solver 999

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information
of ANSYS, Inc. and its subsidiaries and affiliates.

Using This Manual

This preface is divided into the following sections:

1. The Contents of This Manual
2. Typographical Conventions
3. Mathematical Conventions

1. The Contents of This Manual

The Ansys Fluent Theory Guide provides you with theoretical information about the models used in Ansys Fluent.

Important:

Under U.S. and international copyright law, Ansys, Inc. is unable to distribute copies of the papers listed in the bibliography, other than those published internally by Ansys, Inc. Use your library or a document delivery service to obtain copies of copyrighted papers.

A brief description of what is in each chapter follows:

- Basic Fluid Flow (p. 1), describes the governing equations and physical models used by Ansys Fluent to compute fluid flow (including periodic flow, swirling and rotating flows, compressible flows, and inviscid flows).
- Flows with Moving Reference Frames (p. 19), describes single moving reference frames, multiple moving reference frames, and mixing planes.
- Flows Using Sliding and Dynamic Meshes (p. 35), describes sliding and deforming meshes.
- Turbulence (p. 41), describes various turbulent flow models.
- Heat Transfer (p. 171), describes the physical models used to compute heat transfer (including convective and conductive heat transfer, natural convection, radiative heat transfer, and periodic heat transfer).
- Heat Exchangers (p. 223), describes the physical models used to simulate the performance of heat exchangers.
- Species Transport and Finite-Rate Chemistry (p. 237), describes the finite-rate chemistry models. This chapter also provides information about modeling species transport in non-reacting flows.
- Non-Premixed Combustion (p. 295), describes the non-premixed combustion model.
- Premixed Combustion (p. 332), describes the premixed combustion model.
- Partially Premixed Combustion (p. 343), describes the partially premixed combustion model.
- Composition PDF Transport (p. 276), describes the composition PDF transport model.
- Chemistry Acceleration (p. 283), describes the methods used to accelerate computations for detailed chemical mechanisms involving laminar and turbulent flames.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

- Engine Ignition (p. 419), describes the engine ignition models.
- Pollutant Formation (p. 363), describes the models for the formation of NOx and soot.
- Aerodynamically Generated Noise (p. 431), describes the acoustics model.
- Discrete Phase (p. 447), describes the discrete phase models.
- Modeling Macroscopic Particles (p. 589), describes the macroscopic particle model.
- Multiphase Flows (p. 595), describes the general multiphase models (VOF, mixture, and Eulerian).
- Population Balance Model (p. 785), describes the population balance model.
- Solidification and Melting (p. 817), describes the solidification and melting model.
- The Structural Model for Intrinsic Fluid-Structure Interaction (FSI) (p. 827), describes the structural model.
- Eulerian Wall Films (p. 839), describes the Eulerian wall film model.
- Electric Potential and Electrochemistry Models (p. 857), describes the electric potential model.
- Battery Model (p. 871), describes the battery models.
- Modeling Fuel Cells (p. 899), describes the fuel cell modules.
- Modeling Magnetohydrodynamics (p. 935), describes the methods for flow in an electromagnetic field.
- Modeling Continuous Fibers (p. 939), describes the continuous fiber model.
- Solver Theory (p. 955), describes the Fluent solvers.
- Adapting the Mesh (p. 1021), describes the solution-adaptive mesh refinement feature.
- Reporting Alphanumeric Data (p. 1031), describes how to obtain reports of fluxes, forces, surface integrals, and other solution data.

2. Typographical Conventions

Several typographical conventions are used in this manual's text to help you find commands in the user interface.

- Different type styles are used to indicate graphical user interface items and text interface items. For example:

  Iso-Surface dialog box

  `surface/iso-surface` text command

- The text interface type style is also used when illustrating exactly what appears on the screen to distinguish it from the narrative text. In this context, user inputs are typically shown in boldface. For example,

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

```
solve/initialize/set-fmg-initialization

 Customize your FMG initialization:
   set the number of multigrid levels [5]

   set FMG parameters on levels ..

    residual reduction on level 1 is:  [0.001]
    number of cycles on level 1 is:  [10]  100

    residual reduction on level 2 is:  [0.001]
    number of cycles on level 2 is:  [50]  100
```

- Mini flow charts are used to guide you through the ribbon or the tree, leading you to a specific option, dialog box, or task page. The following tables list the meaning of each symbol in the mini flow charts.

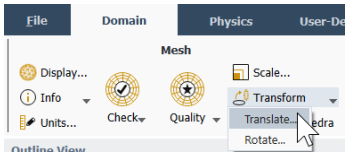
Table 1: Mini Flow Chart Symbol Descriptions

Symbol	Indicated Action
	Look at the ribbon
	Look at the tree
	Double-click to open task page
	Select from task page
	Right-click the preceding item

For example,

Domain → Mesh → Transform → Translate...

indicates selecting the **Domain** ribbon tab, clicking **Transform** (in the **Mesh** group box) and selecting **Translate...**, as indicated in the figure below:

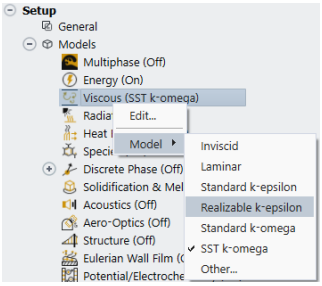


And

Setup → Models → Viscous **Model → Realizable k-epsilon**

indicates expanding the **Setup** and **Models** branches, right-clicking **Viscous**, and selecting **Realizable k-epsilon** from the **Model** sub-menu, as shown in the following figure:

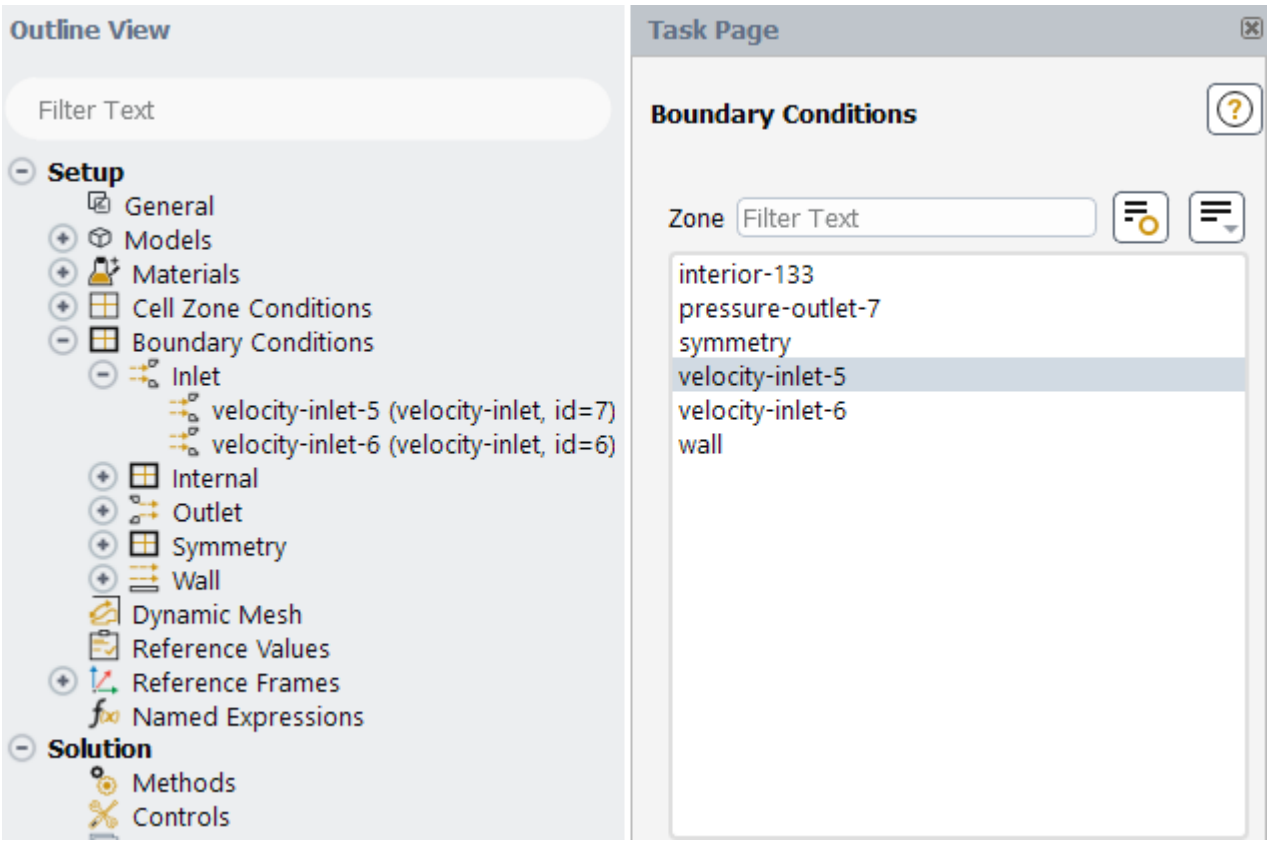
Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.



And

Setup → Boundary Conditions → velocity-inlet-5

indicates opening the task page as shown below:



In this manual, mini flow charts usually accompany a description of a dialog box or command, or a screen illustration showing how to use the dialog box or command. They show you how to quickly access a command or dialog box without having to search the surrounding material.

- In-text references to **File** ribbon tab selections can be indicated using a "/". For example **File/Write/Case...** indicates clicking the **File** ribbon tab and selecting **Case...** from the **Write** submenu (which opens the **Select File** dialog box).

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

3. Mathematical Conventions

- Where possible, vector quantities are displayed with a raised arrow (for example, $\vec{a}$, $\vec{A}$). Boldfaced characters are reserved for vectors and matrices as they apply to linear algebra (for example, the identity matrix, $\mathbf{I}$).

- The operator ∇, referred to as grad, nabla, or del, represents the partial derivative of a quantity with respect to all directions in the chosen coordinate system. In Cartesian coordinates, ∇ is defined to be

$$\frac{\partial}{\partial x}\vec{i}+\frac{\partial}{\partial y}\vec{j}+\frac{\partial}{\partial z}\vec{k} \tag{1}$$

  ∇ appears in several ways:

  – The gradient of a scalar quantity is the vector whose components are the partial derivatives; for example,

$$\nabla p=\frac{\partial p}{\partial x}\vec{i}+\frac{\partial p}{\partial y}\vec{j}+\frac{\partial p}{\partial z}\vec{k} \tag{2}$$

  – The gradient of a vector quantity is a second-order tensor; for example, in Cartesian coordinates,

$$\nabla\left(\vec{v}\right)=\left(\frac{\partial}{\partial x}\vec{i}+\frac{\partial}{\partial y}\vec{j}+\frac{\partial}{\partial z}\vec{k}\right)\left(v_x\vec{i}+v_j\vec{j}+v_z\vec{k}\right) \tag{3}$$

    This tensor is usually written as

$$\begin{pmatrix}\frac{\partial v_x}{\partial x} & \frac{\partial v_x}{\partial y} & \frac{\partial v_x}{\partial z}\\ \frac{\partial v_y}{\partial x} & \frac{\partial v_y}{\partial y} & \frac{\partial v_y}{\partial z}\\ \frac{\partial v_z}{\partial x} & \frac{\partial v_z}{\partial y} & \frac{\partial v_z}{\partial z}\end{pmatrix} \tag{4}$$

  – The divergence of a vector quantity, which is the inner product between ∇ and a vector; for example,

$$\nabla\cdot\vec{v}=\frac{\partial v_x}{\partial x}+\frac{\partial v_y}{\partial y}+\frac{\partial v_z}{\partial z} \tag{5}$$

  – The operator $\nabla\cdot\nabla$, which is usually written as ∇^2 and is known as the Laplacian; for example,

$$\nabla^2 T=\frac{\partial^2 T}{\partial x^2}+\frac{\partial^2 T}{\partial y^2}+\frac{\partial^2 T}{\partial z^2} \tag{6}$$

    $\nabla^2 T$ is different from the expression $\left(\nabla T\right)^2$, which is defined as

$$\left(\nabla T\right)^2=\left(\frac{\partial T}{\partial x}\right)^2+\left(\frac{\partial T}{\partial y}\right)^2+\left(\frac{\partial T}{\partial z}\right)^2 \tag{7}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

- An exception to the use of ∇ is found in the discussion of Reynolds stresses in Turbulence in the *Fluent Theory Guide* (p. 41), where convention dictates the use of Cartesian tensor notation. In this chapter, you will also find that some velocity vector components are written as u, v, and w instead of the conventional v with directional subscripts.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Chapter 1: Basic Fluid Flow

This chapter describes the theoretical background for some of the basic physical models that Ansys Fluent provides for fluid flow.

The information in this chapter is presented in the following sections:

1.1. Overview of Physical Models in Ansys Fluent

1.2. Continuity and Momentum Equations

1.3. User-Defined Scalar (UDS) Transport Equations

1.4. Periodic Flows

1.5. Swirling and Rotating Flows

1.6. Compressible Flows

1.7. Inviscid Flows

For more information about:	See
Models for flows in moving zones (including sliding and dynamic meshes)	Flows with Moving Reference Frames (p. 19) and Flows Using Sliding and Dynamic Meshes (p. 35)
Models for turbulence	Turbulence (p. 41)
Models for heat transfer (including radiation)	Heat Transfer (p. 171)
Models for species transport and reacting flows	Species Transport and Finite-Rate Chemistry (p. 237) – Composition PDF Transport (p. 276)
Models for pollutant formation	Pollutant Formation (p. 363)
Models for discrete phase	Discrete Phase (p. 447)
Models for general multiphase	Multiphase Flows (p. 595)
Models for melting and solidification	Solidification and Melting (p. 817)
Models for porous media, porous jumps, and lumped parameter fans and radiators	Cell Zone and Boundary Conditions in the User's Guide.

1.1. Overview of Physical Models in Ansys Fluent

Ansys Fluent provides comprehensive modeling capabilities for a wide range of incompressible and compressible, laminar and turbulent fluid flow problems. Steady-state or transient analyses can be performed. In Ansys Fluent, a broad range of mathematical models for transport phenomena (like heat transfer and chemical reactions) is combined with the ability to model complex geometries. Examples of Ansys Fluent applications include laminar non-Newtonian flows in process equipment; conjugate heat transfer in turbomachinery and automotive engine components; pulverized coal combustion in utility boilers; external aerodynamics; flow through compressors, pumps, and fans; and multiphase flows in bubble columns and fluidized beds.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

To permit modeling of fluid flow and related transport phenomena in industrial equipment and processes, various useful features are provided. These include porous media, lumped parameter (fan and heat exchanger), streamwise-periodic flow and heat transfer, swirl, and moving reference frame models. The moving reference frame family of models includes the ability to model single or multiple reference frames. A time-accurate sliding mesh method, useful for modeling multiple stages in turbomachinery applications, for example, is also provided, along with the mixing plane model for computing time-averaged flow fields.

Another very useful group of models in Ansys Fluent is the set of free surface and multiphase flow models. These can be used for analysis of gas-liquid, gas-solid, liquid-solid, and gas-liquid-solid flows. For these types of problems, Ansys Fluent provides the volume-of-fluid (VOF), mixture, and Eulerian models, as well as the discrete phase model (DPM). The DPM performs Lagrangian trajectory calculations for dispersed phases (particles, droplets, or bubbles), including coupling with the continuous phase. Examples of multiphase flows include channel flows, sprays, sedimentation, separation, and cavitation.

Robust and accurate turbulence models are a vital component of the Ansys Fluent suite of models. The turbulence models provided have a broad range of applicability, and they include the effects of other physical phenomena, such as buoyancy and compressibility. Particular care has been devoted to addressing issues of near-wall accuracy via the use of extended wall functions and zonal models.

Various modes of heat transfer can be modeled, including natural, forced, and mixed convection with or without conjugate heat transfer, porous media, and so on. The set of radiation models and related submodels for modeling participating media are general and can take into account the complications of combustion. A particular strength of Ansys Fluent is its ability to model combustion phenomena using a variety of models, including eddy dissipation and probability density function models. A host of other models that are very useful for reacting flow applications are also available, including coal and droplet combustion, surface reaction, and pollutant formation models.

1.2. Continuity and Momentum Equations

For all flows, Ansys Fluent solves conservation equations for mass and momentum. For flows involving heat transfer or compressibility, an additional equation for energy conservation is solved. For flows involving species mixing or reactions, a species conservation equation is solved or, if the non-premixed combustion model is used, conservation equations for the mixture fraction and its variance are solved. Additional transport equations are also solved when the flow is turbulent.

In this section, the conservation equations for laminar flow in an inertial (non-accelerating) reference frame are presented. The equations that are applicable to moving reference frames are presented in Flows with Moving Reference Frames (p. 19). The conservation equations relevant to heat transfer, turbulence modeling, and species transport will be discussed in the chapters where those models are described.

The Euler equations solved for inviscid flow are presented in Inviscid Flows (p. 15).

For more information, see the following sections:

1.2.1. The Mass Conservation Equation

1.2.2. Momentum Conservation Equations

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

1.2.1. The Mass Conservation Equation

The equation for conservation of mass, or continuity equation, can be written as follows:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot \left(\rho \vec{v} \right) = S_m \tag{1.1}$$

Equation 1.1 (p. 3) is the general form of the mass conservation equation and is valid for incompressible as well as compressible flows. The source S_m is the mass added to the continuous phase from the dispersed second phase (for example, due to vaporization of liquid droplets) and any user-defined sources.

For 2D axisymmetric geometries, the continuity equation is given by

$$\frac{\partial \rho}{\partial t} + \frac{\partial}{\partial x}\left(\rho v_x \right) + \frac{\partial}{\partial r}\left(\rho v_r \right) + \frac{\rho v_r}{r} = S_m \tag{1.2}$$

where x is the axial coordinate, r is the radial coordinate, v_x is the axial velocity, and v_r is the radial velocity.

1.2.2. Momentum Conservation Equations

Conservation of momentum in an inertial (non-accelerating) reference frame is described by [47] (p. 1055)

$$\frac{\partial}{\partial t}\left(\rho \vec{v} \right) + \nabla \cdot \left(\rho \vec{v} \vec{v} \right) = -\nabla p + \nabla \cdot \left(\bar{\bar{\tau}} \right) + \rho \vec{g} + \vec{F} \tag{1.3}$$

where p is the static pressure, $\bar{\bar{\tau}}$ is the stress tensor (described below), and $\rho \vec{g}$ and $\vec{F}$ are the gravitational body force and external body forces (for example, that arise from interaction with the dispersed phase), respectively. $\vec{F}$ also contains other model-dependent source terms such as porous-media and user-defined sources.

The stress tensor $\bar{\bar{\tau}}$ is given by

$$\bar{\bar{\tau}} = \mu \left[\left(\nabla \vec{v} + \nabla \vec{v}^T \right) - \frac{2}{3} \nabla \cdot \vec{v} I \right] \tag{1.4}$$

where μ is the molecular viscosity, I is the unit tensor, and the second term on the right hand side is the effect of volume dilation.

For 2D axisymmetric geometries, the axial and radial momentum conservation equations are given by

$$\begin{aligned} \frac{\partial}{\partial t}\left(\rho v_x \right) + \frac{1}{r}\frac{\partial}{\partial x}\left(r \rho v_x v_x \right) + \frac{1}{r}\frac{\partial}{\partial r}\left(r \rho v_r v_x \right) = & -\frac{\partial p}{\partial x} \\ & + \frac{1}{r}\frac{\partial}{\partial x}\left[r\mu \left(2\frac{\partial v_x}{\partial x} - \frac{2}{3}\left(\nabla \cdot \vec{v} \right) \right) \right] \\ & + \frac{1}{r}\frac{\partial}{\partial r}\left[r\mu \left(\frac{\partial v_x}{\partial r} + \frac{\partial v_r}{\partial x} \right) \right] + F_x \end{aligned} \tag{1.5}$$

and

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$\begin{aligned}\frac{\partial}{\partial t}\left(\rho v_r\right)+\frac{1}{r}\frac{\partial}{\partial x}\left(r\rho v_x v_r\right)+\frac{1}{r}\frac{\partial}{\partial r}\left(r\rho v_r v_r\right)= & -\frac{\partial p}{\partial r} \\ & +\frac{1}{r}\frac{\partial}{\partial x}\left[r\mu\left(\frac{\partial v_r}{\partial x}+\frac{\partial v_x}{\partial r}\right)\right] \\ & +\frac{1}{r}\frac{\partial}{\partial r}\left[r\mu\left(2\frac{\partial v_r}{\partial r}-\frac{2}{3}\left(\nabla\cdot\vec{v}\right)\right)\right] \\ & -2\mu\frac{v_r}{r^2}+\frac{2}{3}\frac{\mu}{r}\left(\nabla\cdot\vec{v}\right)+\rho\frac{v_z^2}{r}+F_r\end{aligned} \tag{1.6}$$

where

$$\nabla\cdot\vec{v}=\frac{\partial v_x}{\partial x}+\frac{\partial v_r}{\partial r}+\frac{v_r}{r} \tag{1.7}$$

and v_z is the swirl velocity. (See Swirling and Rotating Flows (p. 9) for information about modeling axisymmetric swirl.)

1.3. User-Defined Scalar (UDS) Transport Equations

Ansys Fluent can solve the transport equation for an arbitrary, user-defined scalar (UDS) in the same way that it solves the transport equation for a scalar such as species mass fraction. Extra scalar transport equations may be needed in certain types of combustion applications or for example in plasma-enhanced surface reaction modeling.

This section provides information on how you can specify user-defined scalar (UDS) transport equations to enhance the standard features of Ansys Fluent. Ansys Fluent allows you to define additional scalar transport equations in your model in the User-Defined Scalars Dialog Box. For more information about setting up user-defined scalar transport equations in Ansys Fluent, see User-Defined Scalar (UDS) Transport Equations in the User's Guide.

Information in this section is organized in the following subsections:

1.3.1. Single Phase Flow

1.3.2. Multiphase Flow

1.3.1. Single Phase Flow

For an arbitrary scalar ϕ_k, Ansys Fluent solves the equation

$$\frac{\partial\rho\phi_k}{\partial t}+\frac{\partial}{\partial x_i}\left(\rho u_i\phi_k-\Gamma_k\frac{\partial\phi_k}{\partial x_i}\right)=S_{\phi_k}\quad k=1,\ldots,N \tag{1.8}$$

where Γ_k and S_{ϕ_k} are the diffusion coefficient and source term you supplied for each of the N scalar equations. Note that Γ_k is defined as a tensor in the case of anisotropic diffusivity. The diffusion term is therefore $\nabla\cdot\left(\Gamma_k\cdot\phi_k\right)$

For isotropic diffusivity, Γ_k could be written as $\Gamma_k I$ where I is the identity matrix.

For the steady-state case, Ansys Fluent will solve one of the three following equations, depending on the method used to compute the convective flux:

- If convective flux is *not* to be computed, Ansys Fluent will solve the equation

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$-\frac{\partial}{\partial x_i}\left(\Gamma_k \frac{\partial \phi_k}{\partial x_i}\right)=S_{\phi_k} \quad k=1,\ldots,N \tag{1.9}$$

where Γ_k and S_{ϕ_k} are the diffusion coefficient and source term you supplied for each of the N scalar equations.

- If convective flux is to be computed with mass flow rate, Ansys Fluent will solve the equation

$$\frac{\partial}{\partial x_i}\left(\rho u_i \phi_k - \Gamma_k \frac{\partial \phi_k}{\partial x_i}\right)=S_{\phi_k} \quad k=1,\ldots,N \tag{1.10}$$

- It is also possible to specify a user-defined function to be used in the computation of convective flux. In this case, the user-defined mass flux is assumed to be of the form

$$F=\int_S \rho \vec{u}\cdot d\vec{S} \tag{1.11}$$

where $d\vec{S}$ is the face vector area.

1.3.2. Multiphase Flow

For multiphase flows, Ansys Fluent solves transport equations for two types of scalars: *per phase* and *mixture*. For an arbitrary k scalar in *phase-1*, denoted by ϕ_l^k, Ansys Fluent solves the transport equation inside the volume occupied by *phase-l*

$$\frac{\partial \alpha_l \rho_l \phi_l^k}{\partial t} + \nabla \cdot \left(\alpha_l \rho_l \vec{u}_l \phi_l^k - \alpha_l \Gamma_l^k \nabla \phi_l^k\right)=S_l^k \quad k=1,\ldots,N \tag{1.12}$$

where α_l, ρ_l, and $\vec{u}_l$ are the volume fraction, physical density, and velocity of *phase-l*, respectively. Γ_l^k and S_l^k are the diffusion coefficient and source term, respectively, which you will need to specify. In this case, scalar ϕ_l^k is associated only with one phase (*phase-l*) and is considered an individual field variable of *phase-l*.

The mass flux for *phase-l* is defined as

$$F_l=\int_S \alpha_l \rho_l \vec{u}_l \cdot d\vec{S} \tag{1.13}$$

If the transport variable described by scalar ϕ_l^k represents the physical field that is shared between phases, or is considered the same for each phase, then you should consider this scalar as being associated with a mixture of phases, ϕ^k. In this case, the generic transport equation for the scalar is

$$\frac{\partial \rho_m \phi^k}{\partial t} + \nabla \cdot \left(\rho_m \vec{u}_m \phi^k - \Gamma_m^k \nabla \phi^k\right)=S^{k_m} \quad k=1,\ldots,N \tag{1.14}$$

where mixture density ρ_m, mixture velocity $\vec{u}_m$, and mixture diffusivity for the scalar k Γ_m^k are calculated according to

$$\rho_m=\sum_l \alpha_l \rho_l \tag{1.15}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$\rho_m \vec{u}_m = \sum_l \alpha_l \rho_l \vec{u}_l \tag{1.16}$$

$$F_m = \int_S \rho_m \vec{u}_m \cdot d\vec{S} \tag{1.17}$$

$$\Gamma_m^k = \sum_l \alpha_l \Gamma_l^k \tag{1.18}$$

$$S_m^k = \sum_l S_l^k \tag{1.19}$$

To calculate mixture diffusivity, you will need to specify individual diffusivities for each material associated with individual phases.

Note that if the user-defined mass flux option is activated, then mass fluxes shown in Equation 1.13 (p. 5) and Equation 1.17 (p. 6) will need to be replaced in the corresponding scalar transport equations.

1.4. Periodic Flows

Periodic flow occurs when the physical geometry of interest and the expected pattern of the flow/thermal solution have a periodically repeating nature. Two types of periodic flow can be modeled in Ansys Fluent. In the first type, no pressure drop occurs across the periodic planes. In the second type, a pressure drop occurs across translationally periodic boundaries, resulting in "fully-developed" or "streamwise-periodic" flow.

This section discusses streamwise-periodic flow. A description of no-pressure-drop periodic flow is provided in Periodic Boundary Conditions in the *Fluent User's Guide*, and a description of streamwise-periodic heat transfer is provided in Modeling Periodic Heat Transfer in the *Fluent User's Guide*. For more information about setting up periodic flows in Ansys Fluent, see Periodic Flows in the *Fluent User's Guide*.

Information about streamwise-periodic flow is presented in the following sections:

1.4.1. Overview

1.4.2. Limitations

1.4.3. Physics of Periodic Flows

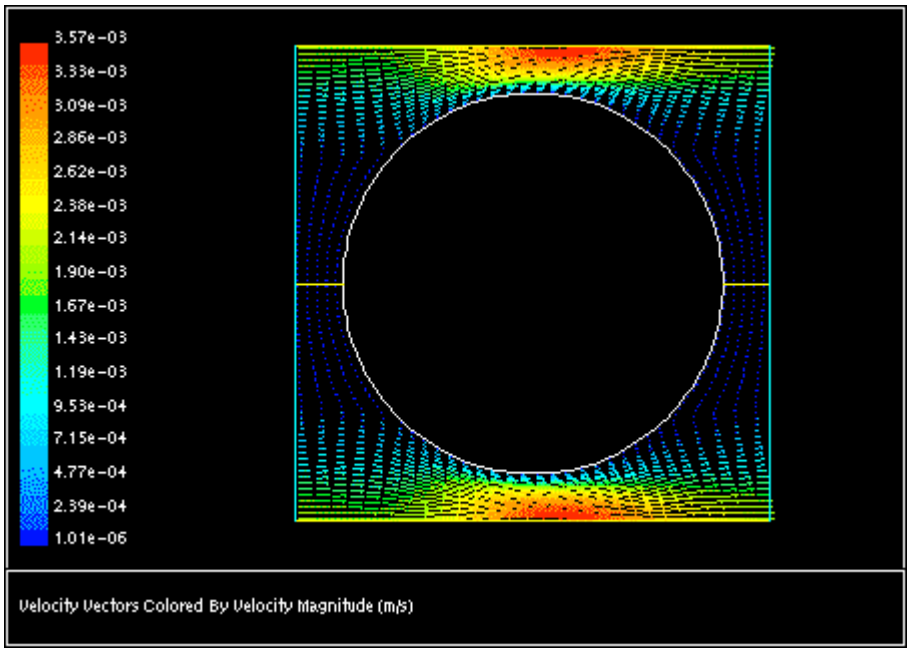
1.4.1. Overview

Ansys Fluent provides the ability to calculate streamwise-periodic — or "fully-developed" — fluid flow. These flows are encountered in a variety of applications, including flows in compact heat exchanger channels and flows across tube banks. In such flow configurations, the geometry varies in a repeating manner along the direction of the flow, leading to a periodic fully-developed flow regime in which the flow pattern repeats in successive cycles. Other examples of streamwise-periodic flows include fully-developed flow in pipes and ducts. These periodic conditions are achieved after a sufficient entrance length, which depends on the flow Reynolds number and geometric configuration.

Streamwise-periodic flow conditions exist when the flow pattern repeats over some length L, with a constant pressure drop across each repeating module along the streamwise direction. Figure 1.1: Example of Periodic Flow in a 2D Heat Exchanger Geometry (p. 7) depicts one example of a periodically repeating flow of this type that has been modeled by including a single representative module.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Figure 1.1: Example of Periodic Flow in a 2D Heat Exchanger Geometry



1.4.2. Limitations

For a list of the limitations that apply when modeling streamwise-periodic flow, see Limitations for Modeling Streamwise-Periodic Flow in the *Fluent User's Guide*; if modeling heat transfer as part of such a simulation, see also Constraints for Periodic Heat Transfer Predictions in the *Fluent User's Guide*.

1.4.3. Physics of Periodic Flows

1.4.3.1. Definition of the Periodic Velocity

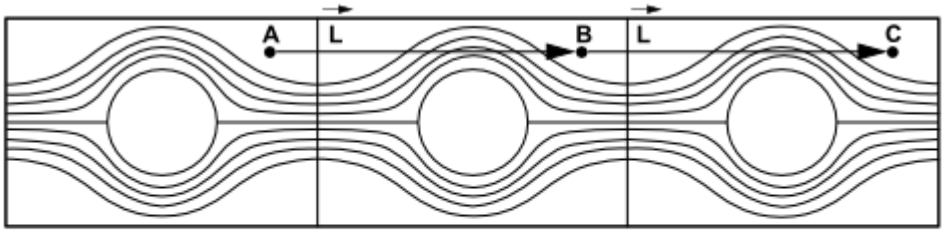
The assumption of periodicity implies that the velocity components repeat themselves in space as follows:

$$\begin{aligned} u(\vec{r}) &= u(\vec{r}+\vec{L}) = u(\vec{r}+2\vec{L}) = \cdots \\ v(\vec{r}) &= v(\vec{r}+\vec{L}) = v(\vec{r}+2\vec{L}) = \cdots \\ w(\vec{r}) &= w(\vec{r}+\vec{L}) = w(\vec{r}+2\vec{L}) = \cdots \end{aligned} \tag{1.20}$$

where $\vec{r}$ is the position vector and $\vec{L}$ is the periodic length vector of the domain considered (see Figure 1.2: Example of a Periodic Geometry (p. 8)).

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Figure 1.2: Example of a Periodic Geometry



$u_A = u_B = u_C$ $\tilde{P}_A = \tilde{P}_B = \tilde{P}_C$

$v_A = v_B = v_C$ $P_B - P_A = P_C - P_B$

1.4.3.2. Definition of the Streamwise-Periodic Pressure

For viscous flows, the pressure is not periodic in the sense of Equation 1.20 (p. 7). Instead, the *pressure drop* between modules is periodic:

$$\Delta p = p(\vec{r}) - p(\vec{r}+\vec{L}) = p(\vec{r}+\vec{L}) - p(\vec{r}+2\vec{L}) = \cdots \tag{1.21}$$

If one of the density-based solvers is used, Δp is specified as a constant value. For the pressure-based solver, the local pressure gradient can be decomposed into two parts: the gradient of a periodic component, $\nabla \tilde{p}(\vec{r})$, and the gradient of a linearly-varying component, $\beta \frac{\vec{L}}{|\vec{L}|}$:

$$\nabla p(\vec{r}) = \beta \frac{\vec{L}}{|\vec{L}|} + \nabla \tilde{p}(\vec{r}) \tag{1.22}$$

where $\tilde{p}(\vec{r})$ is the periodic pressure and $\beta|\vec{r}|$ is the linearly-varying component of the pressure. The periodic pressure is the pressure left over after subtracting out the linearly-varying pressure. The linearly-varying component of the pressure results in a force acting on the fluid in the momentum equations. Because the value of β is not known a priori, it must be iterated on until the mass flow rate that you have defined is achieved in the computational model. This correction of β occurs in the pressure correction step of the SIMPLE, SIMPLEC, or PISO algorithm where the value of β is updated based on the difference between the desired mass flow rate and the actual one. You have some control over the number of sub-iterations used to update β. For more information about setting up parameters for β in Ansys Fluent, see Setting Parameters for the Calculation of β in the *Fluent User's Guide*.

Note:

Because streamwise-periodic flows are "fully developed", the resulting pressure gradient at convergence will only consist of it's linear component. Therefore, the calculated pressure field will not represent a physical pressure.

Integration of Equation 1.21 (p. 8), gives the *static pressure* for translational periodic flow as:

$$p(\vec{r}) = p(\vec{r}_{ref}) + \frac{\beta}{L}\vec{L}\cdot(\vec{r}-\vec{r}_{ref}) + \tilde{p}(\vec{r}) \tag{1.23}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

1.5. Swirling and Rotating Flows

Many important engineering flows involve swirl or rotation and Ansys Fluent is well-equipped to model such flows. Swirling flows are common in combustion, with swirl introduced in burners and combustors in order to increase residence time and stabilize the flow pattern. Rotating flows are also encountered in turbomachinery, mixing tanks, and a variety of other applications.

When you begin the analysis of a rotating or swirling flow, it is essential that you classify your problem into one of the following five categories of flow:

- axisymmetric flows with swirl or rotation
- fully three-dimensional swirling or rotating flows
- flows requiring a moving reference frame
- flows requiring multiple moving reference frames or mixing planes
- flows requiring sliding meshes

Modeling and solution procedures for the first two categories are presented in this section. The remaining three, which all involve "moving zones", are discussed in Flows with Moving Reference Frames (p. 19).

Information about rotating and swirling flows is provided in the following subsections:

1.5.1. Overview of Swirling and Rotating Flows
1.5.2. Physics of Swirling and Rotating Flows

For more information about setting up swirling and rotating flows in Ansys Fluent, see Swirling and Rotating Flows in the *Fluent User's Guide*.

1.5.1. Overview of Swirling and Rotating Flows

1.5.1.1. Axisymmetric Flows with Swirl or Rotation

You can solve a 2D axisymmetric problem that includes the prediction of the circumferential or swirl velocity. The assumption of axisymmetry implies that there are no circumferential gradients in the flow, but that there may be nonzero circumferential velocities. Examples of axisymmetric flows involving swirl or rotation are depicted in Figure 1.3: Rotating Flow in a Cavity (p. 10) and Figure 1.4: Swirling Flow in a Gas Burner (p. 10).

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Figure 1.3: Rotating Flow in a Cavity

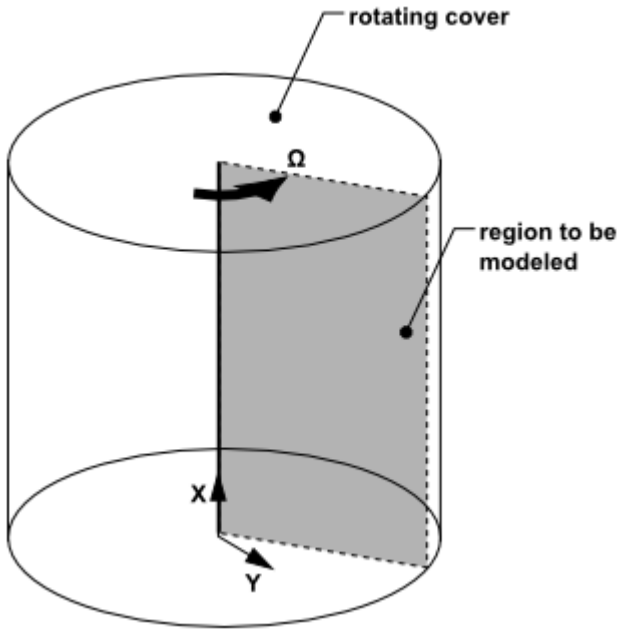
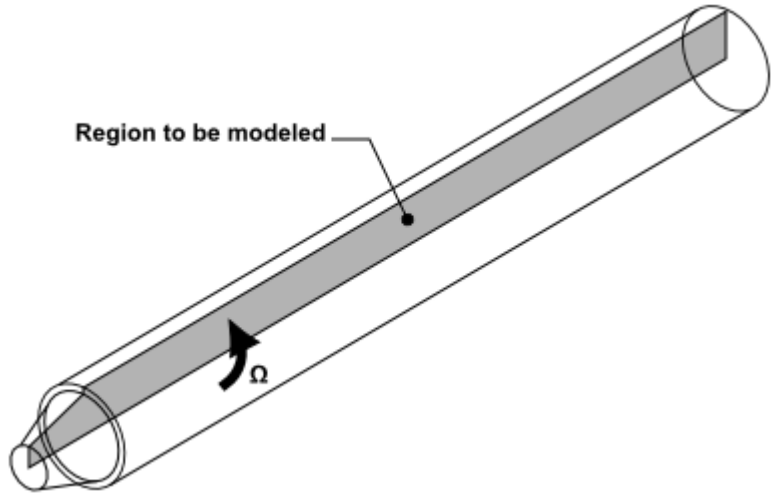


Figure 1.4: Swirling Flow in a Gas Burner



Your problem may be axisymmetric with respect to geometry and flow conditions but still include swirl or rotation. In this case, you can model the flow in 2D (that is, solve the axisymmetric problem) and include the prediction of the circumferential (or swirl) velocity. It is important to note that while the assumption of axisymmetry implies that there are no circumferential gradients in the flow, there may still be nonzero swirl velocities.

1.5.1.1.1. Momentum Conservation Equation for Swirl Velocity

The tangential momentum equation for 2D swirling flows may be written as

$$\frac{\partial}{\partial t}(\rho w)+\frac{1}{r}\frac{\partial}{\partial x}(r\rho uw)+\frac{1}{r}\frac{\partial}{\partial r}(r\rho vw) = \frac{1}{r}\frac{\partial}{\partial x}\left[r\mu\frac{\partial w}{\partial x}\right] + \frac{1}{r^2}\frac{\partial}{\partial r}\left[r^3\mu\frac{\partial}{\partial r}\left(\frac{w}{r}\right)\right] - \rho\frac{vw}{r} \tag{1.24}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

where x is the axial coordinate, r is the radial coordinate, u is the axial velocity, v is the radial velocity, and w is the swirl velocity.

1.5.1.2. Three-Dimensional Swirling Flows

When there are geometric changes and/or flow gradients in the circumferential direction, your swirling flow prediction requires a three-dimensional model. If you are planning a 3D Ansys Fluent model that includes swirl or rotation, you should be aware of the setup constraints (Coordinate System Restrictions in the *Fluent User's Guide*). In addition, you may want to consider simplifications to the problem which might reduce it to an equivalent axisymmetric problem, especially for your initial modeling effort. Because of the complexity of swirling flows, an initial 2D study, in which you can quickly determine the effects of various modeling and design choices, can be very beneficial.

Important:

For 3D problems involving swirl or rotation, there are no special inputs required during the problem setup and no special solution procedures. Note, however, that you may want to use the cylindrical coordinate system for defining velocity-inlet boundary condition inputs, as described in Defining the Velocity in the User's Guide. Also, you may find the gradual increase of the rotational speed (set as a wall or inlet boundary condition) helpful during the solution process. For more information, see Improving Solution Stability by Gradually Increasing the Rotational or Swirl Speed in the User's Guide.

1.5.1.3. Flows Requiring a Moving Reference Frame

If your flow involves a rotating boundary that moves through the fluid (for example, an impeller blade or a grooved or notched surface), you will need to use a moving reference frame to model the problem. Such applications are described in detail in Flow in a Moving Reference Frame (p. 21). If you have more than one rotating boundary (for example, several impellers in a row), you can use multiple reference frames (described in The Multiple Reference Frame Model (p. 24)) or mixing planes (described in The Mixing Plane Model (p. 28)).

1.5.2. Physics of Swirling and Rotating Flows

In swirling flows, conservation of angular momentum (rw or $r^2\Omega$ = constant) tends to create a free vortex flow, in which the circumferential velocity, w, increases sharply as the radius, r, decreases (with w finally decaying to zero near $r=0$ as viscous forces begin to dominate). A tornado is one example of a free vortex. Figure 1.5: Typical Radial Distribution of Circumferential Velocity in a Free Vortex (p. 11) depicts the radial distribution of w in a typical free vortex.

Figure 1.5: Typical Radial Distribution of Circumferential Velocity in a Free Vortex

It can be shown that for an ideal free vortex flow, the centrifugal forces created by the circumferential motion are in equilibrium with the radial pressure gradient:

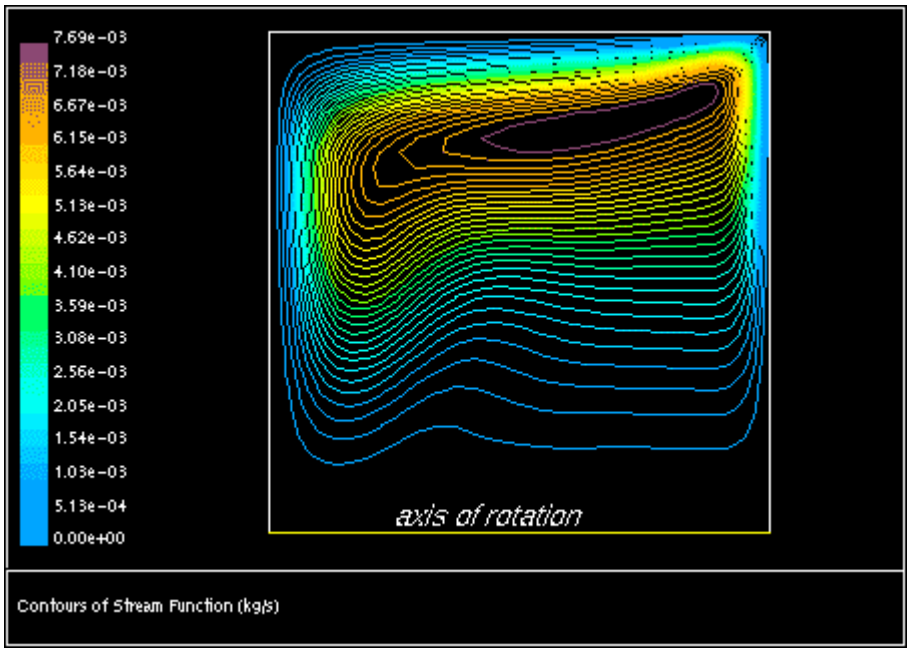
Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$\frac{\partial p}{\partial r} = \frac{\rho w^2}{r} \tag{1.25}$$

As the distribution of angular momentum in a non-ideal vortex evolves, the form of this radial pressure gradient also changes, driving radial and axial flows in response to the highly non-uniform pressures that result. Thus, as you compute the distribution of swirl in your Ansys Fluent model, you will also notice changes in the static pressure distribution and corresponding changes in the axial and radial flow velocities. It is this high degree of coupling between the swirl and the pressure field that makes the modeling of swirling flows complex.

In flows that are driven by wall rotation, the motion of the wall tends to impart a forced vortex motion to the fluid, wherein w/r or Ω is constant. An important characteristic of such flows is the tendency of fluid with high angular momentum (for example, the flow near the wall) to be flung radially outward (see Figure 1.6: Stream Function Contours for Rotating Flow in a Cavity (p. 12) using the geometry of Figure 1.3: Rotating Flow in a Cavity (p. 10)). This is often referred to as "radial pumping", since the rotating wall is pumping the fluid radially outward.

Figure 1.6: Stream Function Contours for Rotating Flow in a Cavity



1.6. Compressible Flows

Compressibility effects are encountered in gas flows at high velocity and/or in which there are large pressure variations. When the flow velocity approaches or exceeds the speed of sound of the gas or when the pressure change in the system ($\Delta p/p$) is large, the variation of the gas density with pressure has a significant impact on the flow velocity, pressure, and temperature. Compressible flows create a unique set of flow physics for which you must be aware of the special input requirements and solution techniques described in this section. Figure 1.7: Transonic Flow in a Converging-Diverging Nozzle (p. 13) and Figure 1.8: Mach 0.675 Flow Over a Bump in a 2D Channel (p. 13) show examples of compressible flows computed using Ansys Fluent.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Figure 1.7: Transonic Flow in a Converging-Diverging Nozzle

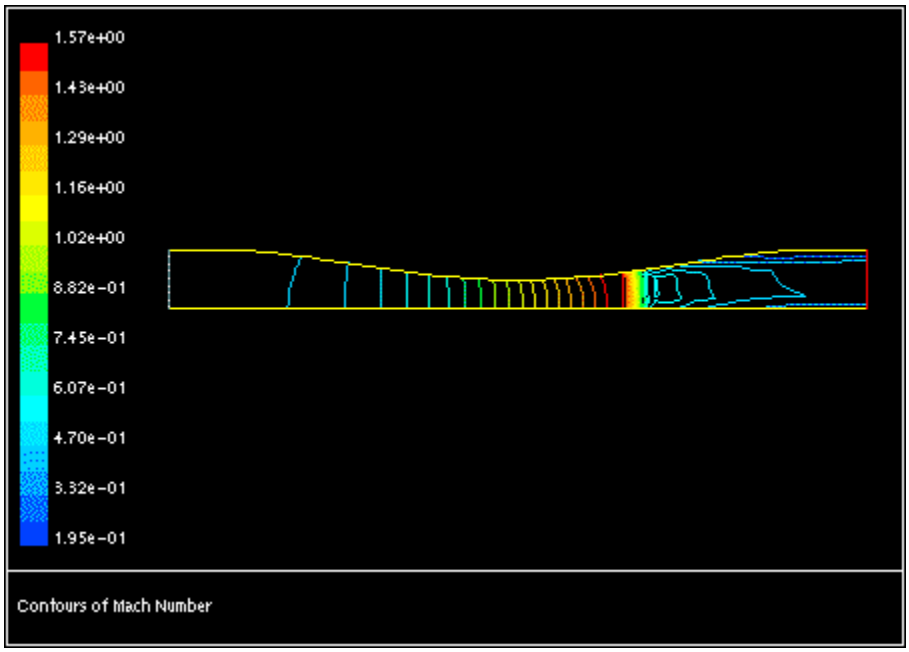
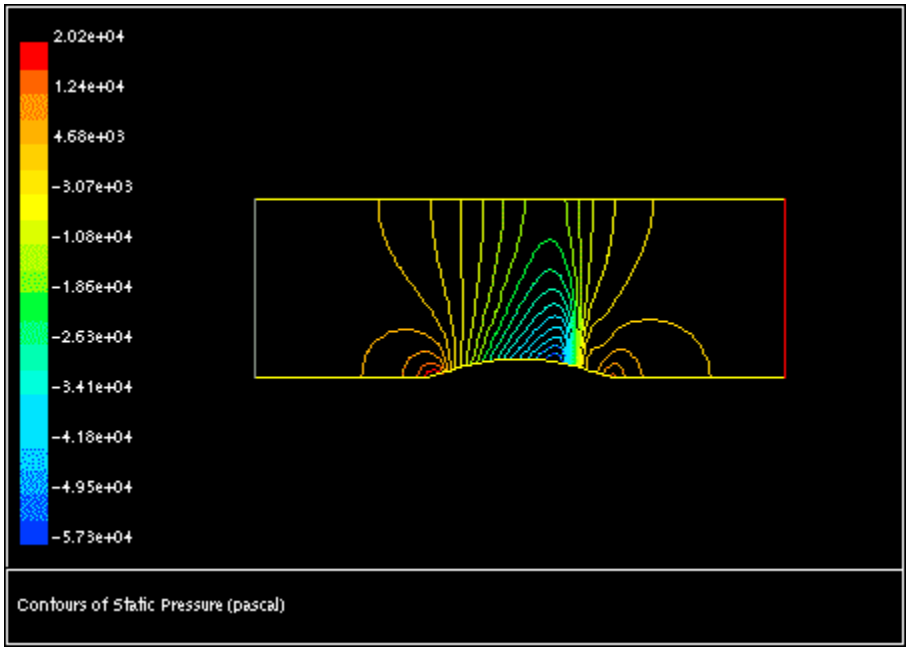


Figure 1.8: Mach 0.675 Flow Over a Bump in a 2D Channel



For more information about setting up compressible flows in Ansys Fluent, see Compressible Flows in the User's Guide.

Information about compressible flows is provided in the following subsections:

1.6.1. When to Use the Compressible Flow Model

1.6.2. Physics of Compressible Flows

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

1.6.1. When to Use the Compressible Flow Model

Compressible flows can be characterized by the value of the Mach number:

$$M \equiv u / c \tag{1.26}$$

Here, c is the speed of sound in the gas:

$$c = \sqrt{\gamma RT} \tag{1.27}$$

and γ is the ratio of specific heats $\left(c_p / c_v\right)$.

Equation 1.27 (p. 14) applies for ideal gas. In the general case of real fluids, the speed of sound is defined in terms of the isentropic compressibility as:

$$c = 1 \Big/ \sqrt{\left(\frac{\partial \rho}{\partial p}\right)_s} \tag{1.28}$$

where ρ is the fluid density, p is the pressure, and the subscript s denotes that the partial derivative of density ρ with respect to pressure p is taken at constant entropy.

When the Mach number is less than 1.0, the flow is termed subsonic. At Mach numbers much less than 1.0 ($M < 0.1$ or so), compressibility effects are negligible and the variation of the gas density with pressure can safely be ignored in your flow modeling. As the Mach number approaches 1.0 (which is referred to as the transonic flow regime), compressibility effects become important. When the Mach number exceeds 1.0, the flow is termed supersonic, and may contain shocks and expansion fans that can impact the flow pattern significantly. Ansys Fluent provides a wide range of compressible flow modeling capabilities for subsonic, transonic, and supersonic flows.

1.6.2. Physics of Compressible Flows

Compressible flows are typically characterized by the total pressure p_0 and total temperature T_0 of the flow. For an ideal gas, these quantities can be related to the static pressure and temperature by the following:

$$\frac{p_0}{p} = exp\left(\frac{\int_T^{T_0} \frac{C_p}{T} dT}{R}\right) \tag{1.29}$$

For constant C_p, Equation 1.29 (p. 14) reduces to

$$\frac{p_0}{p} = \left(1 + \frac{\gamma - 1}{2} M^2\right)^{\gamma/(\gamma - 1)} \tag{1.30}$$

$$\frac{T_0}{T} = 1 + \frac{\gamma - 1}{2} M^2 \tag{1.31}$$

These relationships describe the variation of the static pressure and temperature in the flow as the velocity (Mach number) changes under isentropic conditions. For example, given a pressure ratio from inlet to exit (total to static), Equation 1.30 (p. 14) can be used to estimate the exit Mach number that would exist in a one-dimensional isentropic flow. For air, Equation 1.30 (p. 14) predicts a choked flow (Mach number of 1.0) at an isentropic pressure ratio, p / p_0, of 0.5283. This choked flow condition will be established at the point of minimum flow area (for example, in the throat of a nozzle). In the

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

subsequent area expansion the flow may either accelerate to a supersonic flow in which the pressure will continue to drop, or return to subsonic flow conditions, decelerating with a pressure rise. If a supersonic flow is exposed to an imposed pressure increase, a shock will occur, with a sudden pressure rise and deceleration accomplished across the shock.

1.6.2.1. Basic Equations for Compressible Flows

Compressible flows are described by the standard continuity and momentum equations solved by Ansys Fluent, and you do not need to enable any special physical models (other than the compressible treatment of density as detailed below). The energy equation solved by Ansys Fluent correctly incorporates the coupling between the flow velocity and the static temperature, and should be enabled whenever you are solving a compressible flow. In addition, if you are using the pressure-based solver, you should enable the viscous dissipation terms in Equation 5.1 (p. 172), which become important in high-Mach-number flows.

1.6.2.2. The Compressible Form of the Gas Law

For compressible flows, the ideal gas law is written in the following form:

$$\rho = \frac{p_{op} + p}{\frac{R}{M_w} T} \tag{1.32}$$

where p_{op} is the operating pressure defined in the Operating Conditions Dialog Box, p is the local static pressure relative to the operating pressure, R is the universal gas constant, and M_w is the molecular weight. The temperature, T, will be computed from the energy equation.

Some compressible flow problems involve fluids that do not behave as ideal gases. For example, flow under very high-pressure conditions cannot typically be modeled accurately using the ideal-gas assumption. Therefore, the real gas model described in Real Gas Models in the User's Guide should be used instead.

1.7. Inviscid Flows

Inviscid flow analyses neglect the effect of viscosity on the flow and are appropriate for high-Reynolds-number applications where inertial forces tend to dominate viscous forces. One example for which an inviscid flow calculation is appropriate is an aerodynamic analysis of some high-speed projectile. In a case like this, the pressure forces on the body will dominate the viscous forces. Hence, an inviscid analysis will give you a quick estimate of the primary forces acting on the body. After the body shape has been modified to maximize the lift forces and minimize the drag forces, you can perform a viscous analysis to include the effects of the fluid viscosity and turbulent viscosity on the lift and drag forces.

Another area where inviscid flow analyses are routinely used is to provide a good initial solution for problems involving complicated flow physics and/or complicated flow geometry. In a case like this, the viscous forces are important, but in the early stages of the calculation the viscous terms in the momentum equations will be ignored. Once the calculation has been started and the residuals are decreasing, you can turn on the viscous terms (by enabling laminar or turbulent flow) and continue the solution to convergence. For some very complicated flows, this may be the only way to get the calculation started.

For more information about setting up inviscid flows in Ansys Fluent, see Inviscid Flows in the User's Guide.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Information about inviscid flows is provided in the following section.

1.7.1. Euler Equations

1.7.1. Euler Equations

For inviscid flows, Ansys Fluent solves the Euler equations. The mass conservation equation is the same as for a laminar flow, but the momentum and energy conservation equations are reduced due to the absence of molecular diffusion.

In this section, the conservation equations for inviscid flow in an inertial (non-rotating) reference frame are presented. The equations that are applicable to non-inertial reference frames are described in Flows with Moving Reference Frames (p. 19). The conservation equations relevant for species transport and other models will be discussed in the chapters where those models are described.

1.7.1.1. The Mass Conservation Equation

The equation for conservation of mass, or continuity equation, can be written as follows:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot \left(\rho \vec{v} \right) = S_m \tag{1.33}$$

Equation 1.33 (p. 16) is the general form of the mass conservation equation and is valid for incompressible as well as compressible flows. The source S_m is the mass added to the continuous phase from the dispersed second phase (for example, due to vaporization of liquid droplets) and any user-defined sources.

For 2D axisymmetric geometries, the continuity equation is given by

$$\frac{\partial \rho}{\partial t} + \frac{\partial}{\partial x}\left(\rho v_x \right) + \frac{\partial}{\partial r}\left(\rho v_r \right) + \frac{\rho v_r}{r} = S_m \tag{1.34}$$

where x is the axial coordinate, r is the radial coordinate, v_x is the axial velocity, and v_r is the radial velocity.

1.7.1.2. Momentum Conservation Equations

Conservation of momentum is described by

$$\frac{\partial}{\partial t}\left(\rho \vec{v} \right) + \nabla \cdot \left(\rho \vec{v} \vec{v} \right) = -\nabla p + \rho \vec{g} + \vec{F} \tag{1.35}$$

where p is the static pressure and $\rho \vec{g}$ and $\vec{F}$ are the gravitational body force and external body forces (for example, forces that arise from interaction with the dispersed phase), respectively. $\vec{F}$ also contains other model-dependent source terms such as porous-media and user-defined sources.

For 2D axisymmetric geometries, the axial and radial momentum conservation equations are given by

$$\frac{\partial}{\partial t}\left(\rho v_x \right) + \frac{1}{r}\frac{\partial}{\partial x}\left(r \rho v_x v_x \right) + \frac{1}{r}\frac{\partial}{\partial r}\left(r \rho v_r v_x \right) = -\frac{\partial p}{\partial x} + F_x \tag{1.36}$$

and

$$\frac{\partial}{\partial t}\left(\rho v_r \right) + \frac{1}{r}\frac{\partial}{\partial x}\left(r \rho v_x v_r \right) + \frac{1}{r}\frac{\partial}{\partial r}\left(r \rho v_r v_r \right) = -\frac{\partial p}{\partial r} + F_r \tag{1.37}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

where

$$\nabla \cdot \vec{v} = \frac{\partial v_x}{\partial x} + \frac{\partial v_r}{\partial r} + \frac{v_r}{r} \tag{1.38}$$

1.7.1.3. Energy Conservation Equation

Conservation of energy is described by

$$\frac{\partial}{\partial t}(\rho E) + \nabla \cdot \left(\vec{v}(\rho E + p)\right) = -\nabla \cdot \left(\sum_j h_j J_j\right) + S_h \tag{1.39}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information
of ANSYS, Inc. and its subsidiaries and affiliates.

Chapter 2: Flows with Moving Reference Frames

This chapter describes the theoretical background for modeling flows in moving reference frames. Information about using the various models in this chapter can be found in Modeling Flows with Moving Reference Frames in the User's Guide. The information in this chapter is presented in the following sections:

2.1. Introduction

2.2. Flow in a Moving Reference Frame

2.3. Flow in Multiple Reference Frames

2.1. Introduction

Ansys Fluent solves the equations of fluid flow and heat transfer by default in a stationary (or inertial) reference frame. However, there are many problems where it is advantageous to solve the equations in a moving (or non-inertial) reference frame. These problems typically involve moving parts, such as rotating blades, impellers, and moving walls, and it is the flow around the moving parts that is of interest. In most cases, the moving parts render the problem unsteady when viewed from a stationary frame. With a moving reference frame, however, the flow around the moving part can (with certain restrictions) be modeled as a steady-state problem with respect to the moving frame.

Ansys Fluent's moving reference frame modeling capability allows you to model problems involving moving parts by allowing you to activate moving reference frames in selected cell zones. When a moving reference frame is activated, the equations of motion are modified to incorporate the additional acceleration terms that occur due to the transformation from the stationary to the moving reference frame.

For many problems, it may be possible to refer the entire computational domain to a single moving reference frame (see Figure 2.1: Single Component (Blower Wheel Blade Passage) (p. 20)). This is known as the single reference frame (or SRF) approach. The use of the SRF approach is possible; provided the geometry meets certain requirements (as discussed in Flow in a Moving Reference Frame (p. 21)). For more complex geometries, it may not be possible to use a single reference frame (see Figure 2.2: Multiple Component (Blower Wheel and Casing) (p. 20)). In such cases, you must break up the problem into multiple cell zones, with well-defined interfaces between the zones. The manner in which the interfaces are treated leads to two approximate, steady-state modeling methods for this class of problem: the multiple reference frame (or MRF) approach, and the mixing plane approach. These approaches will be discussed in The Multiple Reference Frame Model (p. 24) and The Mixing Plane Model (p. 28). If unsteady interaction between the stationary and moving parts is important, you can employ the sliding mesh approach to capture the transient behavior of the flow. The sliding meshing model will be discussed in Flows Using Sliding and Dynamic Meshes (p. 35).

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Figure 2.1: Single Component (Blower Wheel Blade Passage)

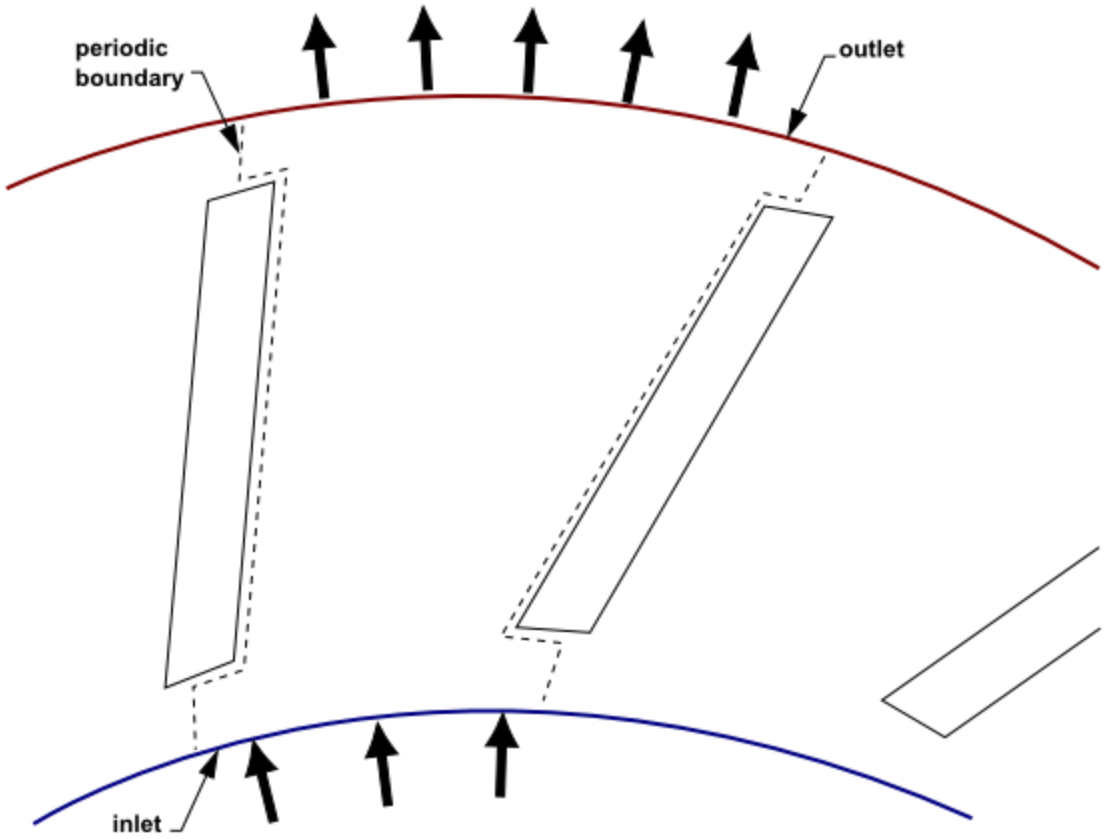
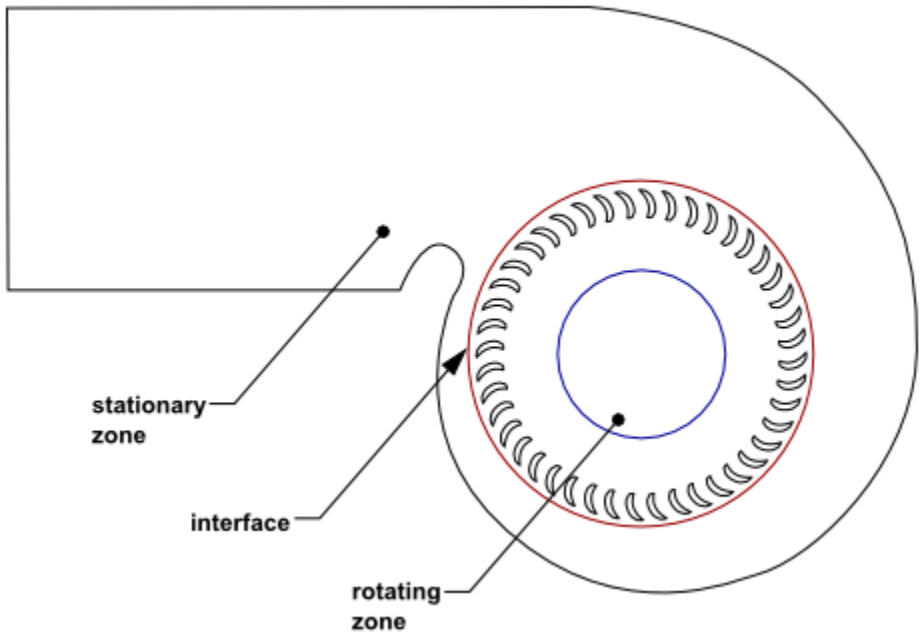


Figure 2.2: Multiple Component (Blower Wheel and Casing)



Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information
of ANSYS, Inc. and its subsidiaries and affiliates.

2.2. Flow in a Moving Reference Frame

The principal reason for employing a moving reference frame is to render a problem that is unsteady in the stationary (inertial) frame steady with respect to the moving frame. For a steadily moving frame (for example, the rotational speed is constant), it is possible to transform the equations of fluid motion to the moving frame such that steady-state solutions are possible. It should also be noted that you can run an unsteady simulation in a moving reference frame with constant rotational speed. This would be necessary if you wanted to simulate, for example, vortex shedding from a rotating fan blade. The unsteadiness in this case is due to a natural fluid instability (vortex generation) rather than induced from interaction with a stationary component.

It is also possible in Ansys Fluent to have frame motion with unsteady translational and rotational speeds. Again, the appropriate acceleration terms are added to the equations of fluid motion. Such problems are inherently unsteady with respect to the moving frame due to the unsteady frame motion

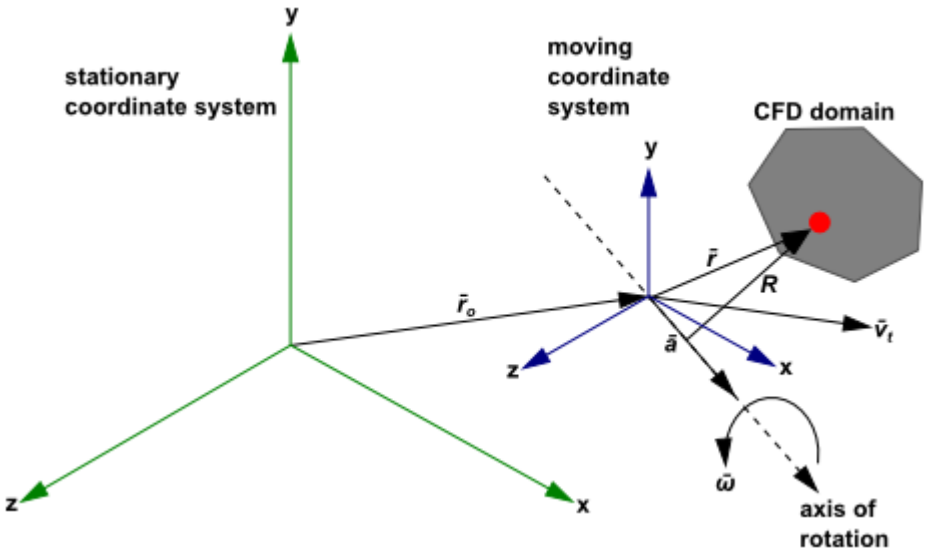
For more information, see the following section:

2.2.1. Equations for a Moving Reference Frame

2.2.1. Equations for a Moving Reference Frame

Consider a coordinate system that is translating with a linear velocity $\vec{v}_t$ and rotating with angular velocity $\vec{\omega}$ relative to a stationary (inertial) reference frame, as illustrated in Figure 2.3: Stationary and Moving Reference Frames (p. 21). The origin of the moving system is located by a position vector $\vec{r}_0$.

Figure 2.3: Stationary and Moving Reference Frames



The axis of rotation is defined by a unit direction vector $\hat{a}$ such that

$$\vec{\omega}=\omega\hat{a} \tag{2.1}$$

The computational domain for the CFD problem is defined with respect to the moving frame such that an arbitrary point in the CFD domain is located by a position vector $\vec{r}$ from the origin of the moving frame.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

The fluid velocities can be transformed from the stationary frame to the moving frame using the following relation:

$$\vec{v}_r = \vec{v} - \vec{u}_r \tag{2.2}$$

where

$$\vec{u}_r = \vec{v}_t + \vec{\omega} \times \vec{r} \tag{2.3}$$

In the above equations, $\vec{v}_r$ is the relative velocity (the velocity viewed from the moving frame), $\vec{v}$ is the absolute velocity (the velocity viewed from the stationary frame), $\vec{u}_r$ is the velocity of the moving frame relative to the inertial reference frame, $\vec{v}_t$ is the translational frame velocity, and $\vec{\omega}$ is the angular velocity. It should be noted that both $\vec{\omega}$ and $\vec{v}_t$ can be functions of time.

When the equations of motion are solved in the moving reference frame, the acceleration of the fluid is augmented by additional terms that appear in the momentum equations [47] (p. 1055). Moreover, the equations can be formulated in two different ways:

- Expressing the momentum equations using the relative velocities as dependent variables (known as the relative velocity formulation).
- Expressing the momentum equations using the absolute velocities as dependent variables in the momentum equations (known as the absolute velocity formulation).

The governing equations for these two formulations will be provided in the sections below. It can be noted here that Ansys Fluent's pressure-based solvers provide the option to use either of these two formulations, whereas the density-based solvers always use the absolute velocity formulation. For more information about the advantages of each velocity formulation, see Choosing the Relative or Absolute Velocity Formulation in the User's Guide.

2.2.1.1. Relative Velocity Formulation

For the relative velocity formulation, the governing equations of fluid flow in a moving reference frame can be written as follows:

Conservation of mass:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot \rho \vec{v}_r = 0 \tag{2.4}$$

Conservation of momentum:

$$\frac{\partial}{\partial t}\left(\rho \vec{v}_r\right) + \nabla \cdot \left(\rho \vec{v}_r \vec{v}_r\right) + \rho\left(2\vec{\omega} \times \vec{v}_r + \vec{\omega} \times \vec{\omega} \times \vec{r} + \vec{\alpha} \times \vec{r} + \vec{a}\right) = -\nabla p + \nabla \cdot \overline{\overline{\tau}}_r + \vec{F} \tag{2.5}$$

where $\vec{\alpha} = \frac{d\vec{\omega}}{dt}$ and $\vec{a} = \frac{d\vec{v}_t}{dt}$

Conservation of energy:

$$\frac{\partial}{\partial t}\left(\rho E_r\right) + \nabla \cdot \left(\rho \vec{v}_r H_r\right) = \nabla \cdot \left(k \nabla T + \overline{\overline{\tau}}_r \cdot \vec{v}_r\right) + S_h \tag{2.6}$$

The momentum equation contains four additional acceleration terms. The first two terms are the Coriolis acceleration ($2\vec{\omega} \times \vec{v}_r$) and the centripetal acceleration ($\vec{\omega} \times \vec{\omega} \times \vec{r}$), respectively. These terms

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

appear for both steadily moving reference frames (that is, and are constant) and accelerating reference frames (that is, and/or are functions of time). The third and fourth terms are due to the unsteady change of the rotational speed and linear velocity, respectively. These terms vanish for constant translation and/or rotational speeds. In addition, the viscous stress ($\overline{\overline{\tau}}_r$) is identical to Equation 1.4 (p. 3) except that relative velocity derivatives are used. The energy equation is written in terms of the relative internal energy (E_r) and the relative total enthalpy (H_r), also known as the rothalpy. These variables are defined as:

$$E_r = h - \frac{p}{\rho} + \frac{1}{2}\left(v_r^2 - u_r^2\right) \tag{2.7}$$

$$H_r = E_r + \frac{p}{\rho} \tag{2.8}$$

2.2.1.2. Absolute Velocity Formulation

For the absolute velocity formulation, the governing equations of fluid flow for a steadily moving frame can be written as follows:

Conservation of mass:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot \rho \vec{v}_r = 0 \tag{2.9}$$

Conservation of momentum:

$$\frac{\partial}{\partial t}\rho\vec{v} + \nabla \cdot (\rho \vec{v}_r \vec{v}) + \rho\left[\vec{\omega} \times \left(\vec{v} - \vec{v}_t\right)\right] = -\nabla p + \nabla \cdot \overline{\overline{\tau}} + \vec{F} \tag{2.10}$$

Conservation of energy:

$$\frac{\partial}{\partial t}\rho E + \nabla \cdot \left(\rho \vec{v}_r H + p\vec{u}_r\right) = \nabla \cdot \left(k\nabla T + \overline{\overline{\tau}} \cdot \vec{v}\right) + S_h \tag{2.11}$$

In this formulation, the Coriolis and centripetal accelerations can be simplified into a single term ($\left[\vec{\omega} \times \left(\vec{v} - \vec{v}_t\right)\right]$). Notice that the momentum equation for the absolute velocity formulation contains no explicit terms involving $\vec{\alpha}$ or $\vec{a}$.

2.2.1.3. Relative Specification of the Reference Frame Motion

Ansys Fluent allows you to specify the frame of motion relative to an already moving (rotating and translating) reference frame. In this case, the resulting velocity vector is computed as

$$\vec{v}_r = \vec{v} - \vec{u}_r \tag{2.12}$$

where

$$\vec{u}_r = \overrightarrow{u_{r1}} + \overrightarrow{u_{r2}} \tag{2.13}$$

and

$$\vec{\omega} = \overrightarrow{\omega_1} + \overrightarrow{\omega_2} \tag{2.14}$$

Equation 2.13 (p. 23) is known as the Galilei transformation.

The rotation vectors are added together as in Equation 2.14 (p. 23), since the motion of the reference frame can be viewed as a solid body rotation, where the rotation rate is constant for every point on the body. In addition, it allows the formulation of the rotation to be an angular velocity axial

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

(also known as pseudo) vector, describing infinitesimal instantaneous transformations. In this case, both rotation rates obey the commutative law. Note that such an approach is not sufficient when dealing with finite rotations. In this case, the formulation of rotation matrices based on Eulerian angles is necessary [541] (p. 1084).

To learn how to specify a moving reference frame within another moving reference frame, refer to Setting Up Multiple Reference Frames in the User's Guide.

2.3. Flow in Multiple Reference Frames

Problems that involve multiple moving parts cannot be modeled with the Single Reference Frame approach. For these problems, you must break up the model into multiple fluid/solid cell zones, with interface boundaries separating the zones. Zones that contain the moving components can then be solved using the moving reference frame equations (Equations for a Moving Reference Frame (p. 21)), whereas stationary zones can be solved with the stationary frame equations. The manner in which the equations are treated at the interface lead to two approaches that are supported in Ansys Fluent:

- Multiple Moving Reference Frames
  - Multiple Reference Frame model (MRF) (see The Multiple Reference Frame Model (p. 24))
  - Mixing Plane Model (MPM) (see The Mixing Plane Model (p. 28))
- Sliding Mesh Model (SMM)

Both the MRF and mixing plane approaches are steady-state approximations, and differ primarily in the manner in which conditions at the interfaces are treated. These approaches will be discussed in the sections below. The sliding mesh model approach is, on the other hand, inherently unsteady due to the motion of the mesh with time. This approach is discussed in Flows Using Sliding and Dynamic Meshes (p. 35).

2.3.1. The Multiple Reference Frame Model

2.3.1.1. Overview

The MRF model [396] (p. 1076) is, perhaps, the simplest of the two approaches for multiple zones. It is a steady-state approximation in which individual cell zones can be assigned different rotational and/or translational speeds. The flow in each moving cell zone is solved using the moving reference frame equations. (For details, see Flow in a Moving Reference Frame (p. 21)). If the zone is stationary (ω=0), the equations reduce to their stationary forms. At the interfaces between cell zones, a local reference frame transformation is performed to enable flow variables in one zone to be used to calculate fluxes at the boundary of the adjacent zone. The MRF interface formulation will be discussed in more detail in The MRF Interface Formulation (p. 26).

It should be noted that the MRF approach does not account for the relative motion of a moving zone with respect to adjacent zones (which may be moving or stationary); the mesh remains fixed for the computation. This is analogous to freezing the motion of the moving part in a specific position and observing the instantaneous flow field with the rotor in that position. Hence, the MRF is often referred to as the "frozen rotor approach."

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

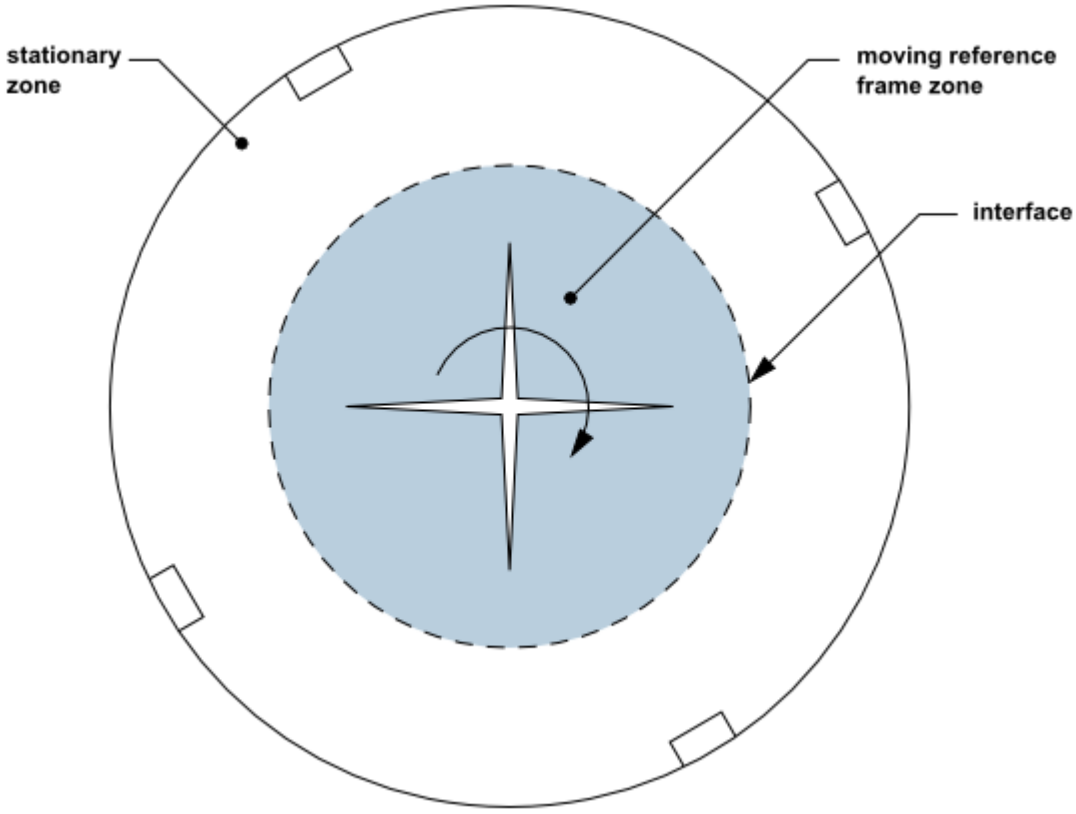
While the MRF approach is clearly an approximation, it can provide a reasonable model of the flow for many applications. For example, the MRF model can be used for turbomachinery applications in which rotor-stator interaction is relatively weak, and the flow is relatively uncomplicated at the interface between the moving and stationary zones. In mixing tanks, since the impeller-baffle interactions are relatively weak, large-scale transient effects are not present and the MRF model can be used.

Another potential use of the MRF model is to compute a flow field that can be used as an initial condition for a transient sliding mesh calculation. This eliminates the need for a startup calculation. The multiple reference frame model should not be used, however, if it is necessary to actually simulate the transients that may occur in strong rotor-stator interactions, as the sliding mesh model alone should be used (see Modeling Flows Using Sliding and Dynamic Meshes in the User's Guide).

2.3.1.2. Examples

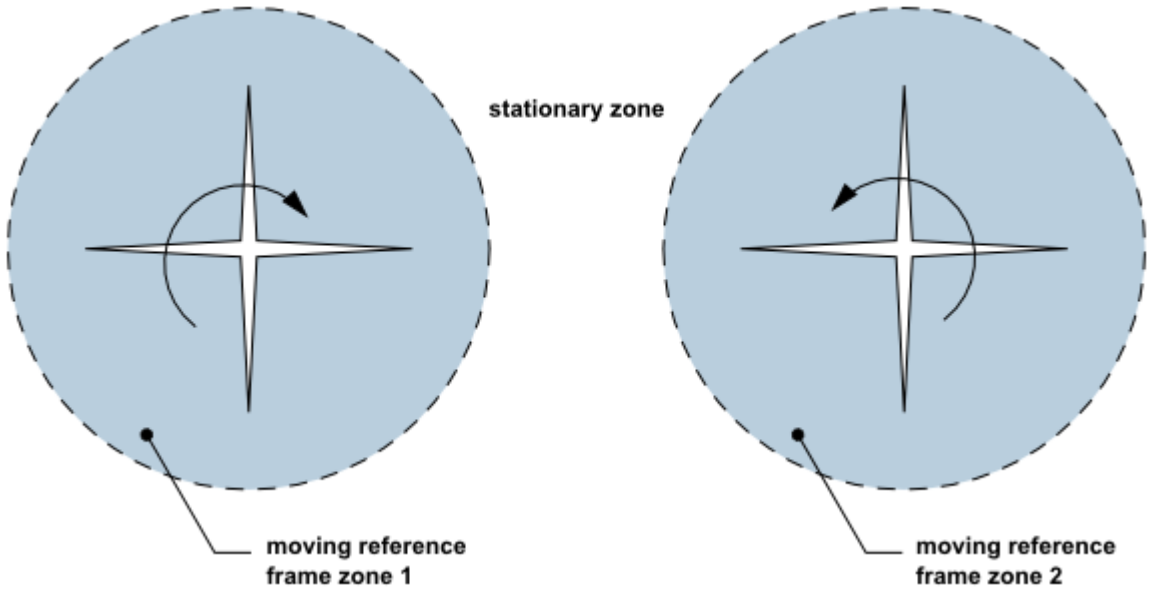
For a mixing tank with a single impeller, you can define a moving reference frame that encompasses the impeller and the flow surrounding it, and use a stationary frame for the flow outside the impeller region. An example of this configuration is illustrated in Figure 2.4: Geometry with One Rotating Impeller (p. 25). (The dashes denote the interface between the two reference frames.) Steady-state flow conditions are assumed at the interface between the two reference frames. That is, the velocity at the interface must be the same (in absolute terms) for each reference frame. *The mesh does not move.*

Figure 2.4: Geometry with One Rotating Impeller



You can also model a problem that includes more than one moving reference frame. Figure 2.5: Geometry with Two Rotating Impellers (p. 26) shows a geometry that contains two rotating impellers side by side. This problem would be modeled using three reference frames: the stationary frame outside both impeller regions and two separate moving reference frames for the two impellers. (As noted above, the dashes denote the interfaces between reference frames.)

Figure 2.5: Geometry with Two Rotating Impellers



2.3.1.3. The MRF Interface Formulation

The MRF formulation that is applied to the interfaces will depend on the velocity formulation being used. The specific approaches will be discussed below for each case. It should be noted that the interface treatment applies to the velocity and velocity gradients, since these vector quantities change with a change in reference frame. Scalar quantities, such as temperature, pressure, density, turbulent kinetic energy, and so on, do not require any special treatment, and therefore are passed locally without any change.

Note:

The interface formulation used by Ansys Fluent does not account for different normal (to the interface) cell zone velocities. You should specify the zone motion of both adjacent cell zones in a way that the interface-normal velocity difference is zero.

2.3.1.3.1. Interface Treatment: Relative Velocity Formulation

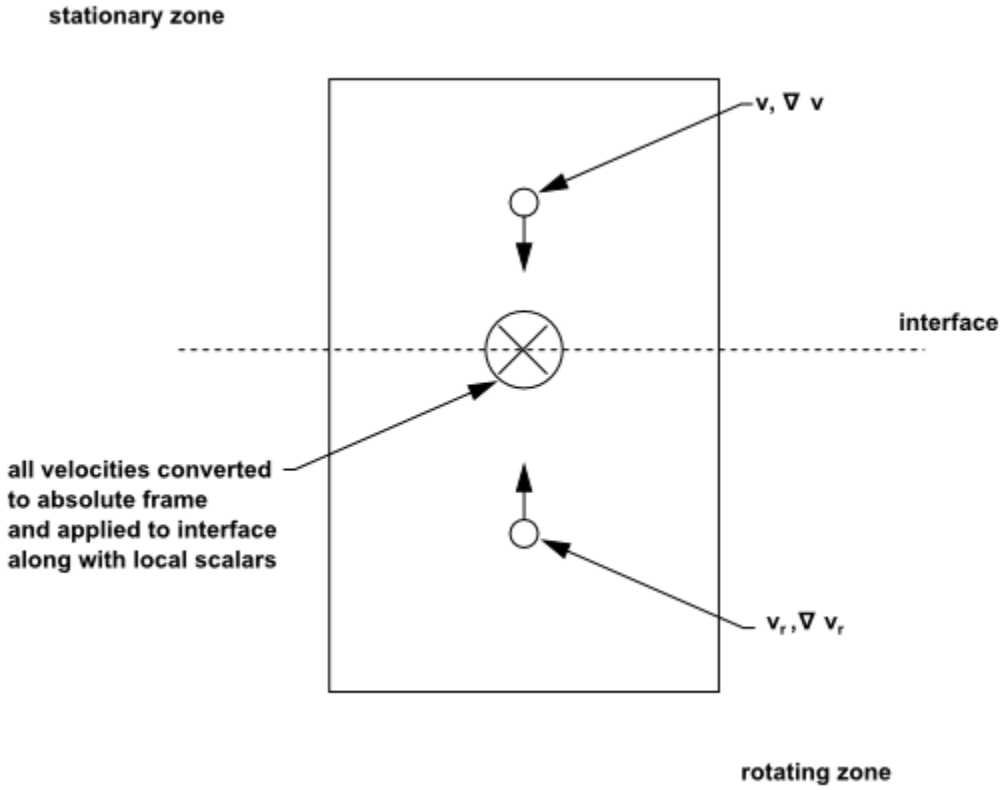
In Ansys Fluent's implementation of the MRF model, the calculation domain is divided into subdomains, each of which may be rotating and/or translating with respect to the laboratory (inertial) frame. The governing equations in each subdomain are written with respect to that subdomain's reference frame. Thus, the flow in stationary and translating subdomains is governed by the equations in Continuity and Momentum Equations (p. 2), while the flow in moving subdomains is governed by the equations presented in Equations for a Moving Reference Frame (p. 21).

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

At the boundary between two subdomains, the diffusion and other terms in the governing equations in one subdomain require values for the velocities in the adjacent subdomain (see Figure 2.6: Interface Treatment for the MRF Model (p. 27)). Ansys Fluent enforces the continuity of the absolute velocity, $\vec{v}$, to provide the correct neighbor values of velocity for the subdomain under consideration. (This approach differs from the mixing plane approach described in The Mixing Plane Model (p. 28), where a circumferential averaging technique is used.)

When the relative velocity formulation is used, velocities in each subdomain are computed relative to the motion of the subdomain. Velocities and velocity gradients are converted from a moving reference frame to the absolute inertial frame using Equation 2.15 (p. 27).

Figure 2.6: Interface Treatment for the MRF Model



For a translational velocity $\vec{v}_t$, we have

$$\vec{v}=\vec{v}_r+\left(\vec{\omega}\times\vec{r}\right)+\vec{v}_t \tag{2.15}$$

From Equation 2.15 (p. 27), the gradient of the absolute velocity vector can be shown to be

$$\nabla\vec{v}=\nabla\vec{v}_r+\nabla\left(\vec{\omega}\times\vec{r}\right) \tag{2.16}$$

Note that scalar quantities such as density, static pressure, static temperature, species mass fractions, and so on, are simply obtained locally from adjacent cells.

2.3.1.3.2. Interface Treatment: Absolute Velocity Formulation

When the absolute velocity formulation is used, the governing equations in each subdomain are written with respect to that subdomain's reference frame, but the velocities are stored in the

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

absolute frame. Therefore, no special transformation is required at the interface between two subdomains. Again, scalar quantities are determined locally from adjacent cells.

2.3.2. The Mixing Plane Model

The mixing plane model in Ansys Fluent provides an alternative to the multiple reference frame and sliding mesh models for simulating flow through domains with one or more regions in relative motion. This section provides a brief overview of the model and a list of its limitations.

2.3.2.1. Overview

As discussed in The Multiple Reference Frame Model (p. 24), the MRF model is applicable when the flow at the interface between adjacent moving/stationary zones is nearly uniform ("mixed out"). If the flow at this interface is not uniform, the MRF model may not provide a physically meaningful solution. The sliding mesh model (see Modeling Flows Using Sliding and Dynamic Meshes in the User's Guide) may be appropriate for such cases, but in many situations it is not practical to employ a sliding mesh. For example, in a multistage turbomachine, if the number of blades is different for each blade row, a large number of blade passages is required in order to maintain circumferential periodicity. Moreover, sliding mesh calculations are necessarily unsteady, and therefore require significantly more computation to achieve a final, time-periodic solution. For situations where using the sliding mesh model is not feasible, the mixing plane model can be a cost-effective alternative.

In the mixing plane approach, each fluid zone is treated as a steady-state problem. Flow-field data from adjacent zones are passed as boundary conditions that are spatially averaged or "mixed" at the mixing plane interface. This mixing removes any unsteadiness that would arise due to circumferential variations in the passage-to-passage flow field (for example, wakes, shock waves, separated flow), therefore yielding a steady-state result. Despite the simplifications inherent in the mixing plane model, the resulting solutions can provide reasonable approximations of the time-averaged flow field.

2.3.2.2. Rotor and Stator Domains

Consider the turbomachine stages shown schematically in Figure 2.7: Axial Rotor-Stator Interaction (Schematic Illustrating the Mixing Plane Concept) (p. 29) and Figure 2.8: Radial Rotor-Stator Interaction (Schematic Illustrating the Mixing Plane Concept) (p. 29), each blade passage contains periodic boundaries. Figure 2.7: Axial Rotor-Stator Interaction (Schematic Illustrating the Mixing Plane Concept) (p. 29) shows a constant radial plane within a single stage of an axial machine, while Figure 2.8: Radial Rotor-Stator Interaction (Schematic Illustrating the Mixing Plane Concept) (p. 29) shows a constant θ plane within a mixed-flow device. In each case, the stage consists of two flow domains: the rotor domain, which is rotating at a prescribed angular velocity, followed by the stator domain, which is stationary. The order of the rotor and stator is arbitrary (that is, a situation where the rotor is downstream of the stator is equally valid).

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Figure 2.7: Axial Rotor-Stator Interaction (Schematic Illustrating the Mixing Plane Concept)

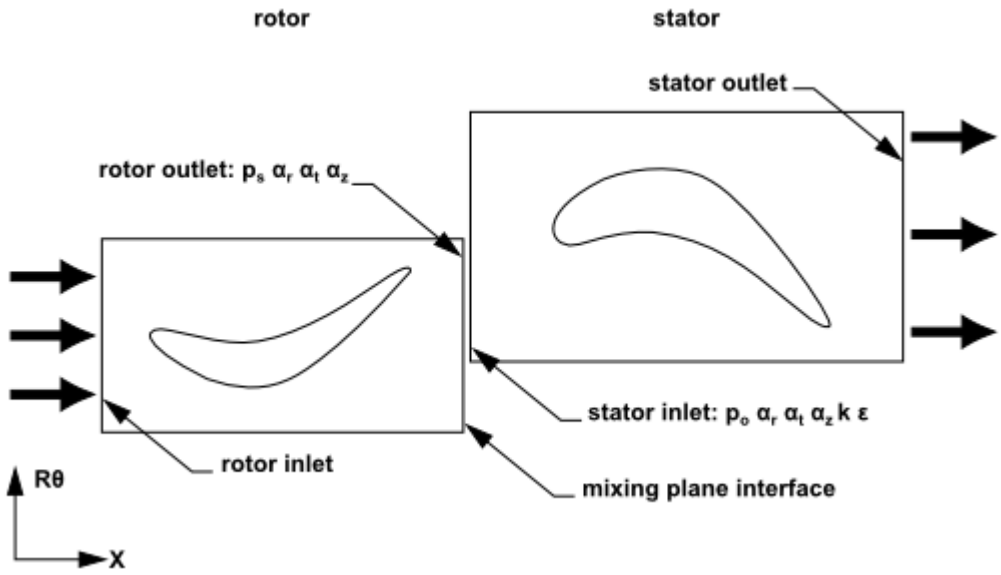
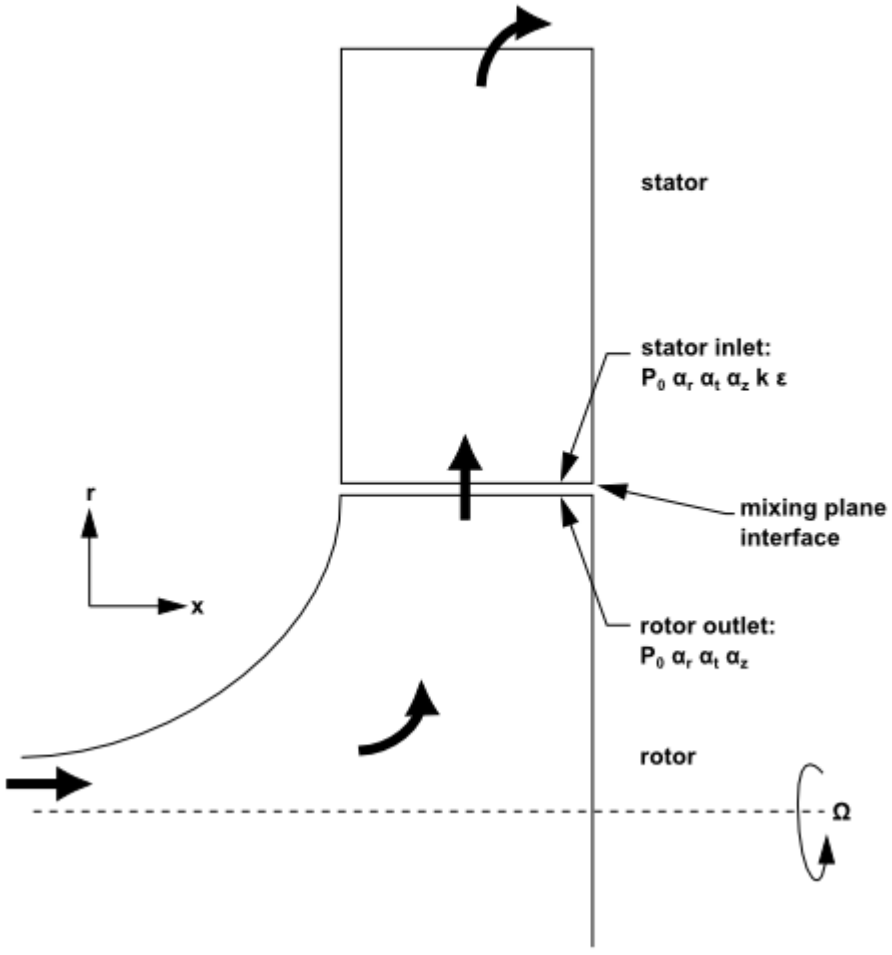


Figure 2.8: Radial Rotor-Stator Interaction (Schematic Illustrating the Mixing Plane Concept)



Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

In a numerical simulation, each domain will be represented by a separate mesh. The flow information between these domains will be coupled at the mixing plane interface (as shown in Figure 2.7: Axial Rotor-Stator Interaction (Schematic Illustrating the Mixing Plane Concept) (p. 29) and Figure 2.8: Radial Rotor-Stator Interaction (Schematic Illustrating the Mixing Plane Concept) (p. 29)) using the mixing plane model. Note that you may couple any number of fluid zones in this manner; for example, four blade passages can be coupled using three mixing planes.

Important:

Note that the stator and rotor passages are separate cell zones, each with their own inlet and outlet boundaries. You can think of this system as a set of SRF models for each blade passage coupled by boundary conditions supplied by the mixing plane model.

2.3.2.3. The Mixing Plane Concept

The essential idea behind the mixing plane concept is that each fluid zone is solved as a steady-state problem. At some prescribed iteration interval, the flow data at the mixing plane interface are averaged in the circumferential direction on both the stator outlet and the rotor inlet boundaries. The Ansys Fluent implementation gives you the choice of three types of averaging methods: area-weighted averaging, mass averaging, and mixed-out averaging. By performing circumferential averages at specified radial or axial stations, "profiles" of boundary condition flow variables can be defined. These profiles—which will be functions of either the axial or the radial coordinate, depending on the orientation of the mixing plane—are then used to update boundary conditions along the two zones of the mixing plane interface. In the examples shown in Figure 2.7: Axial Rotor-Stator Interaction (Schematic Illustrating the Mixing Plane Concept) (p. 29) and Figure 2.8: Radial Rotor-Stator Interaction (Schematic Illustrating the Mixing Plane Concept) (p. 29), profiles of averaged total pressure (p_0), direction cosines of the local flow angles in the radial, tangential, and axial directions ($\alpha_r, \alpha_t, \alpha_z$), total temperature ($T_0$), turbulence kinetic energy (k), and turbulence dissipation rate (ε) are computed at the rotor exit and used to update boundary conditions at the stator inlet. Likewise, a profile of static pressure (p_s), direction cosines of the local flow angles in the radial, tangential, and axial directions ($\alpha_r, \alpha_t, \alpha_z$), are computed at the stator inlet and used as a boundary condition on the rotor exit.

Passing profiles in the manner described above assumes specific boundary condition types have been defined at the mixing plane interface. The coupling of an upstream outlet boundary zone with a downstream inlet boundary zone is called a "mixing plane pair". In order to create mixing plane pairs in Ansys Fluent, the boundary zones must be of the following types:

Upstream	Downstream
pressure outlet	pressure inlet
pressure outlet	velocity inlet
pressure outlet	mass-flow inlet

For specific instructions about setting up mixing planes, see Setting Up the Legacy Mixing Plane Model in the User's Guide.

2.3.2.4. Choosing an Averaging Method

Three profile averaging methods are available in the mixing plane model:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

- area averaging
- mass averaging
- mixed-out averaging

2.3.2.4.1. Area Averaging

Area averaging is the default averaging method and is given by

$$\overline{f} = \frac{1}{A}\int_A f dA \tag{2.17}$$

Important:

The pressure and temperature obtained by the area average may not be representative of the momentum and energy of the flow.

2.3.2.4.2. Mass Averaging

Mass averaging is given by

$$\overline{f} = \frac{1}{\dot{m}}\int_A f \rho \vec{V} \cdot \hat{n} dA \tag{2.18}$$

where

$$\dot{m} = \int_A \rho \vec{V} \cdot \hat{n} dA \tag{2.19}$$

This method provides a better representation of the total quantities than the area-averaging method. Convergence problems could arise if severe reverse flow is present at the mixing plane. Therefore, for solution stability purposes, it is best if you initiate the solution with area averaging, then switch to mass averaging after reverse flow dies out.

Important:

Mass averaging is not available with multiphase flows.

2.3.2.4.3. Mixed-Out Averaging

The mixed-out averaging method is derived from the conservation of mass, momentum and energy:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$
\begin{aligned}
F &= \int_A \rho\left(\vec{V}\cdot\hat{n}\right)dA \\
M_1 &= \int_A \rho\left(\vec{V}\cdot\hat{n}\right)udA + \int_A P\left(\hat{e_x}\cdot\hat{n}\right)dA \\
M_2 &= \int_A \rho\left(\vec{V}\cdot\hat{n}\right)vdA + \int_A P\left(\hat{e_y}\cdot\hat{n}\right)dA \\
M_3 &= \int_A \rho\left(\vec{V}\cdot\hat{n}\right)wdA + \int_A P\left(\hat{e_z}\cdot\hat{n}\right)dA \\
E &= \frac{\gamma R}{(\gamma-1)}\int_A \rho\left(\vec{V}\cdot\hat{n}\right)TdA + \frac{1}{2}\int_A \rho\left(\vec{V}\cdot\hat{n}\right)\left(u^2+v^2+w^2\right)dA
\end{aligned}
\tag{2.20}
$$

Because it is based on the principles of conservation, the mixed-out average is considered a better representation of the flow since it reflects losses associated with non-uniformities in the flow profiles. However, like the mass-averaging method, convergence difficulties can arise when severe reverse flow is present across the mixing plane. Therefore, it is best if you initiate the solution with area averaging, then switch to mixed-out averaging after reverse flow dies out.

Mixed-out averaging assumes that the fluid is a compressible ideal-gas with constant specific heat, C_p.

Important:

Mixed-out averaging is not available with multiphase flows.

2.3.2.5. Mixing Plane Algorithm of Ansys Fluent

Ansys Fluent's basic mixing plane algorithm can be described as follows:

1. Update the flow field solutions in the stator and rotor domains.
2. Average the flow properties at the stator exit and rotor inlet boundaries, obtaining profiles for use in updating boundary conditions.
3. Pass the profiles to the boundary condition inputs required for the stator exit and rotor inlet.
4. Repeat steps 1–3 until convergence.

Important:

Note that it may be desirable to under-relax the changes in boundary condition values in order to prevent divergence of the solution (especially early in the computation). Ansys Fluent allows you to control the under-relaxation of the mixing plane variables.

2.3.2.6. Mass Conservation

Note that the algorithm described above will not rigorously conserve mass flow across the mixing plane if it is represented by a pressure outlet and pressure inlet mixing plane pair. If you use a pressure outlet and mass-flow inlet pair instead, Ansys Fluent will force mass conservation across the mixing plane. The basic technique consists of computing the mass flow rate across the upstream

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

zone (pressure outlet) and adjusting the mass flux profile applied at the mass-flow inlet such that the downstream mass flow matches the upstream mass flow. This adjustment occurs at every iteration, therefore ensuring rigorous conservation of mass flow throughout the course of the calculation.

Important:

Note that, since mass flow is being fixed in this case, there will be a jump in total pressure across the mixing plane. The magnitude of this jump is usually small compared with total pressure variations elsewhere in the flow field.

2.3.2.7. Swirl Conservation

By default, Ansys Fluent does not conserve swirl across the mixing plane. For applications such as torque converters, where the sum of the torques acting on the components should be zero, enforcing swirl conservation across the mixing plane is essential, and is available in Ansys Fluent as a modeling option. Ensuring conservation of swirl is important because, otherwise, sources or sinks of tangential momentum will be present at the mixing plane interface.

Consider a control volume containing a stationary or moving component (for example, a pump impeller or turbine vane). Using the moment of momentum equation from fluid mechanics, it can be shown that for steady flow,

$$T = \int\int_S r v_\theta \rho \vec{v} \cdot \hat{n} dS \tag{2.21}$$

where T is the torque of the fluid acting on the component, r is the radial distance from the axis of rotation, v_θ is the absolute tangential velocity, $\vec{v}$ is the total absolute velocity, and S is the boundary surface. (The product rv_θ is referred to as swirl.)

For a circumferentially periodic domain, with well-defined inlet and outlet boundaries, Equation 2.21 (p. 33) becomes

$$T = \int\int_{outlet} r v_\theta \rho \vec{v} \cdot \hat{n} dS + \int\int_{inlet} r v_\theta \rho \vec{v} \cdot \hat{n} dS \tag{2.22}$$

where inlet and outlet denote the inlet and outlet boundary surfaces.

Now consider the mixing plane interface to have a finite streamwise thickness. Applying Equation 2.22 (p. 33) to this zone and noting that, in the limit as the thickness shrinks to zero, the torque should vanish, the equation becomes

$$\int\int_{downstream} r v_\theta \rho \vec{v} \cdot \hat{n} dS = \int\int_{upstream} r v_\theta \rho \vec{v} \cdot \hat{n} dS \tag{2.23}$$

where upstream and downstream denote the upstream and downstream sides of the mixing plane interface. Note that Equation 2.23 (p. 33) applies to the full area (360 degrees) at the mixing plane interface.

Equation 2.23 (p. 33) provides a rational means of determining the tangential velocity component. That is, Ansys Fluent computes a profile of tangential velocity and then uniformly adjusts the profile such that the swirl integral is satisfied. Note that interpolating the tangential (and radial) velocity

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

component profiles at the mixing plane does not affect mass conservation because these velocity components are orthogonal to the face-normal velocity used in computing the mass flux.

2.3.2.8. Total Enthalpy Conservation

By default, Ansys Fluent does not conserve total enthalpy across the mixing plane. For some applications, total enthalpy conservation across the mixing plane is very desirable, because global parameters such as efficiency are directly related to the change in total enthalpy across a blade row or stage. This is available in Ansys Fluent as a modeling option.

The procedure for ensuring conservation of total enthalpy simply involves adjusting the downstream total temperature profile such that the integrated total enthalpy matches the upstream integrated total enthalpy. For multiphase flows, conservation of mass, swirl, and enthalpy are calculated for each phase. However, for the Eulerian multiphase model, since mass-flow inlets are not permissible, conservation of the above quantities does not occur.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Chapter 3: Flows Using Sliding and Dynamic Meshes

This chapter describes the theoretical background of the sliding and dynamic mesh models in Ansys Fluent. To learn more about using sliding meshes in Ansys Fluent, see Using Sliding Meshes in the User's Guide. For more information about using dynamic meshes in Ansys Fluent, see Using Dynamic Meshes in the User's Guide.

Theoretical information about sliding and dynamic mesh models is presented in the following sections:

3.1. Introduction

3.2. Dynamic Mesh Theory

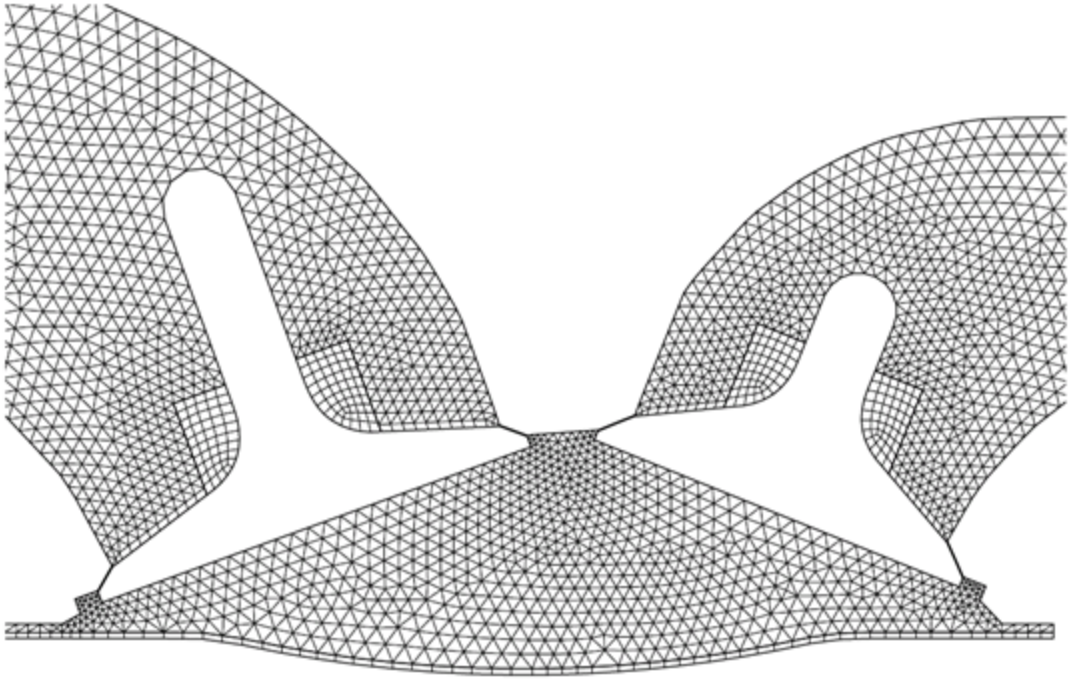
3.3. Sliding Mesh Theory

3.1. Introduction

The dynamic mesh model allows you to move the boundaries of a cell zone relative to other boundaries of the zone, and to adjust the mesh accordingly. The motion of the boundaries can be *rigid*, such as pistons moving inside an engine cylinder (see Figure 3.1: A Mesh Associated With Moving Pistons (p. 36)) or a flap deflecting on an aircraft wing, or *deforming*, such as the elastic wall of a balloon during inflation or a flexible artery wall responding to the pressure pulse from the heart. In either case, the nodes that define the cells in the domain must be updated as a function of time, and hence the dynamic mesh solutions are inherently unsteady. The governing equations describing the fluid motion (which are different from those used for moving reference frames, as described in Flows with Moving Reference Frames (p. 19)) are described in Dynamic Mesh Theory (p. 37).

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

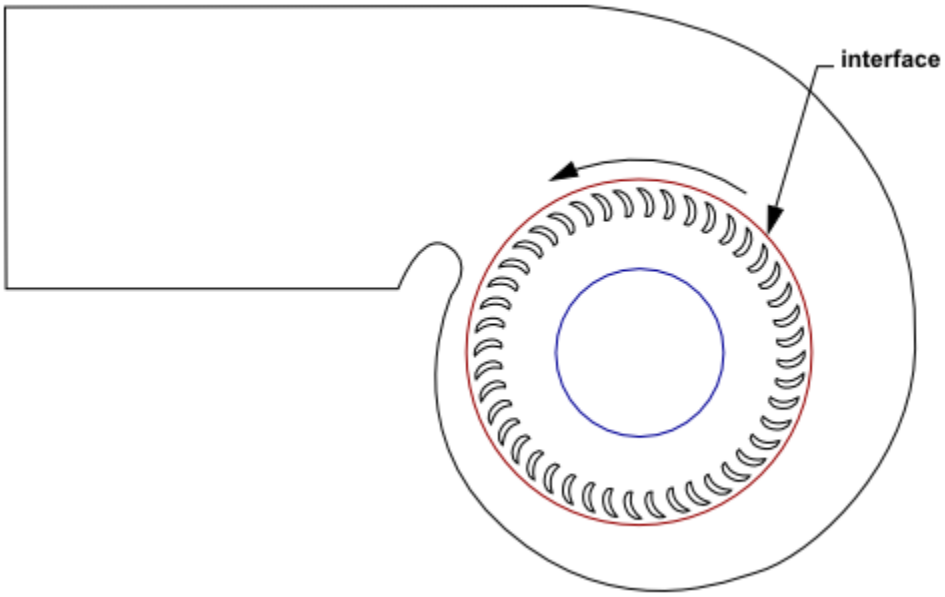
Figure 3.1: A Mesh Associated With Moving Pistons



An important special case of dynamic mesh motion is called the sliding mesh in which all of the boundaries and the cells of a given mesh zone move together in a rigid-body motion. In this situation, the nodes of the mesh move in space (relative to the fixed, global coordinates), but the cells defined by the nodes do not deform. Furthermore, mesh zones moving adjacent to one another can be linked across one or more non-conformal interfaces. As long as the interfaces stay in contact with one another (that is, "slide" along a common overlap boundary at the interface), the non-conformal interfaces can be dynamically updated as the meshes move, and fluid can pass from one zone to the other. Such a scenario is referred to as the sliding mesh model in Ansys Fluent. Examples of sliding mesh model usage include modeling rotor-stator interaction between a moving blade and a stationary vane in a compressor or turbine, modeling a blower with rotating blades and a stationary casing (see Figure 3.2: Blower (p. 37)), and modeling a train moving in a tunnel by defining sliding interfaces between the train and the tunnel walls.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Figure 3.2: Blower



3.2. Dynamic Mesh Theory

The dynamic mesh model in Ansys Fluent can be used to model flows where the shape of the domain is changing with time due to motion on the domain boundaries. The dynamic mesh model can be applied to single or multiphase flows (and multi-species flows). The generic transport equation (Equation 3.1 (p. 38)) applies to all applicable model equations, such as turbulence, energy, species, phases, and so on. The dynamic mesh model can also be used for steady-state applications, when it is beneficial to move the mesh in the steady-state solver. The motion can be a prescribed motion (for example, you can specify the linear and angular velocities about the center of gravity of a solid body with time) or an unprescribed motion where the subsequent motion is determined based on the solution at the current time (for example, the linear and angular velocities are calculated from the force balance on a solid body, as is done by the six degree of freedom (six DOF) solver; see Six DOF Solver Settings in the User's Guide). The update of the volume mesh is handled automatically by Ansys Fluent at each time step based on the new positions of the boundaries. To use the dynamic mesh model, you need to provide a starting volume mesh and the description of the motion of any moving zones in the model. Ansys Fluent allows you to describe the motion using either boundary profiles, user-defined functions (UDFs), or the six degree of freedom solver.

Ansys Fluent expects the description of the motion to be specified on either face or cell zones. If the model contains moving and non-moving regions, you need to identify these regions by grouping them into their respective face or cell zones in the starting volume mesh that you generate. Furthermore, regions that are deforming due to motion on their adjacent regions must also be grouped into separate zones in the starting volume mesh. The boundary between the various regions need not be conformal. You can use the non-conformal or sliding interface capability in Ansys Fluent to connect the various zones in the final model.

Information about dynamic mesh theory is presented in the following sections:

3.2.1. Conservation Equations

3.2.2. Six DOF Solver Theory

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

3.2.1. Conservation Equations

With respect to dynamic meshes, the integral form of the conservation equation for a general scalar, ϕ, on an arbitrary control volume, V, whose boundary is moving can be written as

$$\frac{d}{dt}\int_V \rho\phi dV + \int_{\partial V} \rho\phi\left(\vec{u}-\vec{u}_g\right)\cdot d\vec{A} = \int_{\partial V} \Gamma\nabla\phi\cdot d\vec{A} + \int_V S_\phi dV \tag{3.1}$$

where

ρ is the fluid density

$\vec{u}$ is the flow velocity vector

$\vec{u}_g$ is the mesh velocity of the moving mesh

Γ is the diffusion coefficient

S_ϕ is the source term of ϕ

Here, ∂V is used to represent the boundary of the control volume, V.

By using a first-order backward difference formula, the time derivative term in Equation 3.1 (p. 38) can be written as

$$\frac{d}{dt}\int_V \rho\phi dV = \frac{(\rho\phi V)^{n+1} - (\rho\phi V)^{n}}{\Delta t} \tag{3.2}$$

where n and $n+1$ denote the respective quantity at the current and next time level, respectively. The $(n+1)$th time level volume, V^{n+1}, is computed from

$$V^{n+1} = V^n + \frac{dV}{dt}\Delta t \tag{3.3}$$

where dV/dt is the volume time derivative of the control volume. In order to satisfy the mesh conservation law, the volume time derivative of the control volume is computed from

$$\frac{dV}{dt} = \int_{\partial V} \vec{u}_g\cdot d\vec{A} = \sum_j^{n_f} \vec{u}_{g,j}\cdot\vec{A}_j \tag{3.4}$$

where n_f is the number of faces on the control volume and $\vec{A}_j$ is the j face area vector. The dot product $\vec{u}_{g,j}\cdot\vec{A}_j$ on each control volume face is calculated from

$$\vec{u}_{g,j}\cdot\vec{A}_j = \frac{\delta V_j}{\Delta t} \tag{3.5}$$

where δV_j is the volume swept out by the control volume face j over the time step Δt.

By using a second-order backward difference formula, the time derivative in Equation 3.1 (p. 38) can be written as

$$\frac{d}{dt}\int_V \rho\phi dV = \frac{3(\rho\phi V)^{n+1} - 4(\rho\phi V)^{n} + (\rho\phi V)^{n-1}}{2\Delta t} \tag{3.6}$$

where $n+1$, n, and $n-1$ denote the respective quantities from successive time levels with $n+1$ denoting the current time level.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

In the case of a second-order difference scheme the volume time derivative of the control volume is computed in the same manner as in the first-order scheme as shown in Equation 3.4 (p. 38). For the second-order differencing scheme, the dot product $\vec{u}_{g,j}\cdot\vec{A}_j$ on each control volume face is calculated from

$$\left(\vec{u}_{g,j}\cdot\vec{A}_j\right)^{n+1}=\frac{3}{2}\left(\vec{u}_{g,j}\cdot\vec{A}_j\right)^{n}-\frac{1}{2}\left(\vec{u}_{g,j}\cdot\vec{A}_j\right)^{n-1}=\frac{3}{2}\left(\frac{\delta V_j}{\delta t}\right)^{n}-\frac{1}{2}\left(\frac{\delta V_j}{\delta t}\right)^{n-1} \tag{3.7}$$

where $\left(\delta V_j\right)^n$ and $\left(\delta V_j\right)^{n-1}$ are the volumes swept out by control volume faces at the current and previous time levels over a time step.

3.2.2. Six DOF Solver Theory

The six DOF solver in Ansys Fluent uses the object's forces and moments in order to compute the translational and angular motion of the center of gravity of an object. The governing equation for the translational motion of the center of gravity is solved for in the inertial coordinate system:

$$\dot{\vec{v}}_G=\frac{1}{m}\sum\vec{f}_G \tag{3.8}$$

where $\vec{v}_G$ is the translational motion of the center of gravity, m is the mass, and $\vec{f}_G$ is the force vector due to gravity.

The angular motion of the object, $\dot{\vec{\omega}}_B$, is more easily computed using body coordinates:

$$\dot{\vec{\omega}}_B=L^{-1}\left(\sum\vec{M}_B-\vec{\omega}_B\times L\vec{\omega}_B\right) \tag{3.9}$$

where L is the inertia tensor, $\vec{M}_B$ is the moment vector of the body, and $\vec{\omega}_B$ is the rigid body angular velocity vector.

The moments are transformed from inertial to body coordinates using

$$\vec{M}_B=R\vec{M}_G \tag{3.10}$$

where, R represents the following transformation matrix:

$$\begin{bmatrix} C_\theta C_\psi & C_\theta S_\psi & -S_\theta \\ S_\phi S_\theta C_\psi - C_\phi S_\psi & S_\phi S_\theta S_\psi + C_\phi C_\psi & S_\phi C_\theta \\ C_\phi S_\theta C_\psi + S_\phi S_\psi & C_\phi S_\theta S_\psi - S_\phi C_\psi & C_\phi C_\theta \end{bmatrix}$$

where, in generic terms, $C_\chi=cos\left(\chi\right)$ and $S_\chi=sin\left(\chi\right)$. The angles ϕ, θ, and ψ are Euler angles that represent the following sequence of rotations:

- rotation about the Z axis (for example, yaw for airplanes)
- rotation about the Y axis (for example, pitch for airplanes)
- rotation about the X axis (for example, roll for airplanes)

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

After the angular and the translational accelerations are computed from Equation 3.8 (p. 39) and Equation 3.9 (p. 39), the rates are derived by numerical integration [611] (p. 1088). The angular and translational velocities are used in the dynamic mesh calculations to update the rigid body position.

3.3. Sliding Mesh Theory

As mentioned previously, the sliding mesh model is a special case of general dynamic mesh motion wherein the nodes move rigidly in a given dynamic mesh zone. Additionally, multiple cells zones are connected with each other through non-conformal interfaces. As the mesh motion is updated in time, the non-conformal interfaces are likewise updated to reflect the new positions each zone. It is important to note that the mesh motion must be prescribed such that zones linked through non-conformal interfaces remain in contact with each other (that is, "slide" along the interface boundary) if you want fluid to be able to flow from one mesh to the other. Any portion of the interface where there is no contact is treated as a wall, as described in Non-Conformal Meshes in the User's Guide.

The general conservation equation formulation for dynamic meshes, as expressed in Equation 3.1 (p. 38), is also used for sliding meshes. Because the mesh motion in the sliding mesh formulation is rigid, all cells retain their original shape and volume. As a result, the time rate of change of the cell volume is zero, and Equation 3.3 (p. 38) simplifies to:

$$V^{n+1} = V^{n} \tag{3.11}$$

and Equation 3.2 (p. 38) becomes:

$$\frac{d}{dt}\int_{V} \rho\phi dV = \frac{\left[(\rho\phi)^{n+1} - (\rho\phi)^{n}\right]V}{\Delta t} \tag{3.12}$$

Additionally, Equation 3.4 (p. 38) simplifies to

$$\sum_{j}^{n_f} \vec{u}_{g,j} \cdot \vec{A}_j = 0 \tag{3.13}$$

Equation 3.1 (p. 38), in conjunction with the above simplifications, permits the flow in the moving mesh zones to be updated, provided that an appropriate specification of the rigid mesh motion is defined for each zone (usually this is simple linear or rotation motion, but more complex motions can be used). Note that due to the fact that the mesh is moving, the solutions to Equation 3.1 (p. 38) for sliding mesh applications will be inherently unsteady (as they are for all dynamic meshes).

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Chapter 4: Turbulence

This chapter provides theoretical background about the turbulence models available in Ansys Fluent. Information is presented in the following sections:

4.1. Underlying Principles of Turbulence Modeling

4.2. Spalart-Allmaras Model

4.3. Standard, RNG, and Realizable k-ε Models

4.4. Standard, BSL, and SST k-ω Models

4.5. Generalized k-ω (GEKO) Model

4.6. k-kl-ω Transition Model

4.7. Transition SST Model

4.8. Intermittency Transition Model

4.9. Algebraic Transition Model

4.10. Explicit Algebraic Reynolds Stress Model (EARSM)

4.11. Reynolds Stress Model (RSM)

4.12. Scale-Adaptive Simulation (SAS) Model

4.13. Detached Eddy Simulation (DES)

4.14. Shielded Detached Eddy Simulation (SDES)

4.15. Stress-Blended Eddy Simulation (SBES)

4.16. Large Eddy Simulation (LES) Model

4.17. Embedded Large Eddy Simulation (ELES)

4.18. Near-Wall Treatments for Wall-Bounded Turbulent Flows

4.19. Curvature Correction for the Spalart-Allmaras and Two-Equation Models

4.20. Corner Flow Correction

4.21. Production Limiters for Two-Equation Models

4.22. Turbulence Damping

4.23. Definition of Turbulence Scales

For more information about using these turbulence models in Ansys Fluent, see Modeling Turbulence in the User's Guide.

4.1. Underlying Principles of Turbulence Modeling

The following sections provide an overview of the underlying principles for turbulence modeling.

4.1.1. Reynolds (Ensemble) Averaging

4.1.2. Filtered Navier-Stokes Equations

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

4.1.3. Hybrid RANS-LES Formulations

4.1.4. Boussinesq Approach vs. Reynolds Stress Transport Models

4.1.1. Reynolds (Ensemble) Averaging

In Reynolds averaging, the solution variables in the instantaneous (exact) Navier-Stokes equations are decomposed into the mean (ensemble-averaged or time-averaged) and fluctuating components. For the velocity components:

$$u_i = \bar{u}_i + u_i' \tag{4.1}$$

where $\bar{u}_i$ and u_i' are the mean and fluctuating velocity components ($i = 1,2,3$).

Likewise, for pressure and other scalar quantities:

$$\varphi = \overline{\varphi} + \varphi' \tag{4.2}$$

where φ denotes a scalar such as pressure, energy, or species concentration.

Substituting expressions of this form for the flow variables into the instantaneous continuity and momentum equations and taking a time (or ensemble) average (and dropping the overbar on the mean velocity, $\bar{u}$) yields the ensemble-averaged momentum equations. They can be written in Cartesian tensor form as:

$$\frac{\partial \rho}{\partial t} + \frac{\partial}{\partial x_i}\left(\rho u_i\right) = 0 \tag{4.3}$$

$$\begin{aligned} \frac{\partial}{\partial t}\left(\rho u_i\right) + \frac{\partial}{\partial x_j}\left(\rho u_i u_j\right) = & -\frac{\partial p}{\partial x_i} + \frac{\partial}{\partial x_j}\left[\mu\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} - \frac{2}{3}\delta_{ij}\frac{\partial u_l}{\partial x_l}\right)\right] \\ & + \frac{\partial}{\partial x_j}\left(-\rho\overline{u_i' u_j'}\right) \end{aligned} \tag{4.4}$$

Equation 4.3 (p. 42) and Equation 4.4 (p. 42) are called Reynolds-averaged Navier-Stokes (RANS) equations. They have the same general form as the instantaneous Navier-Stokes equations, with the velocities and other solution variables now representing ensemble-averaged (or time-averaged) values. Additional terms now appear that represent the effects of turbulence. These Reynolds stresses, $-\rho\overline{u_i' u_j'}$, must be modeled in order to close Equation 4.4 (p. 42).

For variable-density flows, Equation 4.3 (p. 42) and Equation 4.4 (p. 42) can be interpreted as Favre-averaged Navier-Stokes equations [246] (p. 1067), with the velocities representing mass-averaged values. As such, Equation 4.3 (p. 42) and Equation 4.4 (p. 42) can be applied to variable-density flows.

4.1.2. Filtered Navier-Stokes Equations

The governing equations employed for LES are obtained by filtering the time-dependent Navier-Stokes equations in either Fourier (wave-number) space or configuration (physical) space. The filtering process effectively filters out the eddies whose scales are smaller than the filter width or grid spacing used in the computations. The resulting equations therefore govern the dynamics of large eddies.

A filtered variable (denoted by an overbar) is defined by Equation 4.5 (p. 42):

$$\overline{\phi}(x) = \int_D \phi(x') G(x, x') dx' \tag{4.5}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

where D is the fluid domain, and G is the filter function that determines the scale of the resolved eddies.

In Ansys Fluent, the finite-volume discretization itself implicitly provides the filtering operation:

$$\overline{\varphi}(x)=\frac{1}{V}\int_{\nu}\varphi(x')dx',x'\in\nu \tag{4.6}$$

where V is the volume of a computational cell. The filter function, $G(x,x')$, implied here is then

$$G(x,x')=\begin{cases}1/\ V, & x'\in\nu\\ 0, & x'otherwise\end{cases} \tag{4.7}$$

The LES capability in Ansys Fluent is applicable to compressible and incompressible flows. For the sake of concise notation, however, the theory that follows is limited to a discussion of incompressible flows.

Filtering the continuity and momentum equations, one obtains

$$\frac{\partial\rho}{\partial t}+\frac{\partial}{\partial x_i}(\rho\overline{u}_i)=0 \tag{4.8}$$

and

$$\frac{\partial}{\partial t}(\rho\overline{u}_i)+\frac{\partial}{\partial x_j}(\rho\overline{u}_i\overline{u}_j)=\frac{\partial}{\partial x_j}(\sigma_{ij})-\frac{\partial\overline{p}}{\partial x_i}-\frac{\partial\tau_{ij}}{\partial x_j} \tag{4.9}$$

where σ_{ij} is the stress tensor due to molecular viscosity defined by

$$\sigma_{ij}\equiv\left[\mu\left(\frac{\partial\overline{u}_i}{\partial x_j}+\frac{\partial\overline{u}_j}{\partial x_i}\right)\right]-\frac{2}{3}\mu\frac{\partial\overline{u}_l}{\partial x_l}\delta_{ij} \tag{4.10}$$

and τ_{ij} is the subgrid-scale stress defined by

$$\tau_{ij}\equiv\rho\overline{u_iu_j}-\rho\overline{u}_i\overline{u}_j \tag{4.11}$$

Filtering the energy equation, one obtains:

$$\frac{\partial\rho\overline{h_s}}{\partial t}+\frac{\partial\rho\overline{u}_i\overline{h_s}}{\partial x_i}-\frac{\partial\overline{p}}{\partial t}-\overline{u_j}\frac{\partial\overline{p}}{\partial x_i}-\frac{\partial}{\partial x_i}\left(\lambda\frac{\partial\overline{T}}{\partial x_i}\right)=-\frac{\partial}{\partial x_j}\left[\underbrace{\rho\left(\overline{u_ih_s}-\overline{u}_i\overline{h_s}\right)}_{\text{subgrid enthalpy flux}}\right] \tag{4.12}$$

where h_s and λ are the sensible enthalpy and thermal conductivity, respectively.

The subgrid enthalpy flux term in the Equation 4.12 (p. 43) is approximated using the gradient hypothesis:

$$\rho\left(\overline{u_ih_s}-\overline{u}_i\overline{h_s}\right)=-\frac{\mu_{\text{SGS}}C_p}{\text{Pr}_{\text{SGS}}}\frac{\partial\overline{T}}{\partial x_j} \tag{4.13}$$

where μ_{SGS} is a subgrid viscosity, and Pr_{SGS} is a subgrid Prandtl number equal to 0.85.

4.1.3. Hybrid RANS-LES Formulations

At first, the concepts of Reynolds Averaging and Spatial Filtering seem incompatible, as they result in different additional terms in the momentum equations (Reynolds Stresses and sub-grid stresses).

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

This would preclude hybrid models like Scale-Adaptive Simulation (SAS), Detached Eddy Simulation (DES), Shielded DES (SDES), or Stress-Blended Eddy Simulation (SBES), which are based on one set of momentum equations throughout the RANS and LES portions of the domain. However, it is important to note that once a turbulence model is introduced into the momentum equations, they no longer carry any information concerning their derivation (averaging). Case in point is that the most popular models, both in RANS and LES, are eddy viscosity models that are used to substitute either the Reynolds- or the sub-grid stress tensor. After the introduction of an eddy viscosity (turbulent viscosity), both the RANS and LES momentum equations are formally identical. The difference lies exclusively in the size of the eddy-viscosity provided by the underlying turbulence model. This allows the formulation of turbulence models that can switch from RANS to LES mode, by lowering the eddy viscosity in the LES zone appropriately, without any formal change to the momentum equations.

4.1.4. Boussinesq Approach vs. Reynolds Stress Transport Models

The Reynolds-averaged approach to turbulence modeling requires that the Reynolds stresses in Equation 4.4 (p. 42) are appropriately modeled. A common method employs the Boussinesq hypothesis [246] (p. 1067) to relate the Reynolds stresses to the mean velocity gradients:

$$-\rho\overline{u_i'u_j'}=\mu_t\left(\frac{\partial u_i}{\partial x_j}+\frac{\partial u_j}{\partial x_i}\right)-\frac{2}{3}\left(\rho k+\mu_t\frac{\partial u_k}{\partial x_k}\right)\delta_{ij} \tag{4.14}$$

The Boussinesq hypothesis is used in the Spalart-Allmaras model, the k-ε models, and the k-ω models. The advantage of this approach is the relatively low computational cost associated with the computation of the turbulent viscosity, μ_t. In the case of the Spalart-Allmaras model, only one additional transport equation (representing turbulent viscosity) is solved. In the case of the k-ε and k-ω models, two additional transport equations (for the turbulence kinetic energy, k, and either the turbulence dissipation rate, ε, or the specific dissipation rate, ω) are solved, and μ_t is computed as a function of k and ε or k and ω. The disadvantage of the Boussinesq hypothesis as presented is that it assumes μ_t is an isotropic scalar quantity, which is not strictly true. However the assumption of an isotropic turbulent viscosity typically works well for shear flows dominated by only one of the turbulent shear stresses. This covers many technical flows, such as wall boundary layers, mixing layers, jets, and so on.

The alternative approach, embodied in the RSM, is to solve transport equations for each of the terms in the Reynolds stress tensor. An additional scale-determining equation (normally for ε or ω) is also required. This means that five additional transport equations are required in 2D flows and seven additional transport equations must be solved in 3D.

In many cases, models based on the Boussinesq hypothesis perform very well, and the additional computational expense of the Reynolds stress model is not justified. However, the RSM is clearly superior in situations where the anisotropy of turbulence has a dominant effect on the mean flow. Such cases include highly swirling flows and stress-driven secondary flows.

4.2. Spalart-Allmaras Model

This section describes the theory behind the Spalart-Allmaras model. Information is presented in the following sections:

4.2.1. Overview

4.2.2. Transport Equation for the Spalart-Allmaras Model

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

4.2.3. Modeling the Turbulent Viscosity
4.2.4. Modeling the Turbulent Production
4.2.5. Modeling the Turbulent Destruction
4.2.6. Model Constants
4.2.7. Wall Boundary Conditions
4.2.8. Convective Heat and Mass Transfer Modeling

For details about using the model in Ansys Fluent, see Modeling Turbulence and Setting Up the Spalart-Allmaras Model in the User's Guide.

4.2.1. Overview

The Spalart-Allmaras model [618] (p. 1089) is a one-equation model that solves a modeled transport equation for the kinematic eddy (turbulent) viscosity. The Spalart-Allmaras model was designed specifically for aerospace applications involving wall-bounded flows and has been shown to give good results for boundary layers subjected to adverse pressure gradients. It is also gaining popularity in turbomachinery applications.

In its original form, the Spalart-Allmaras model is effectively a low-Reynolds number model, requiring the viscosity-affected region of the boundary layer to be properly resolved ($y^+ \sim 1$ meshes). In Ansys Fluent, the Spalart-Allmaras model has been extended with a y^+-insensitive wall treatment, which allows the application of the model independent of the near-wall y^+ resolution. The formulation blends automatically from a viscous sublayer formulation to a logarithmic formulation based on y^+. On intermediate grids, $(1 < y^+ < 30)$, the formulation maintains its integrity and provides consistent wall shear stress and heat transfer coefficients. While the y^+ sensitivity is removed, it still should be ensured that the boundary layer is resolved with a minimum resolution of 10-15 cells.

The Spalart-Allmaras model was developed for aerodynamic flows. It is not calibrated for general industrial flows, and does produce relatively larger errors for some free shear flows, especially plane and round jet flows. In addition, it cannot be relied on to predict the decay of homogeneous, isotropic turbulence.

4.2.2. Transport Equation for the Spalart-Allmaras Model

The transported variable in the Spalart-Allmaras model, $\tilde{\nu}$, is identical to the turbulent kinematic viscosity except in the near-wall (viscosity-affected) region. The transport equation for the modified turbulent viscosity $\tilde{\nu}$ is

$$\frac{\partial}{\partial t}(\rho\tilde{\nu}) + \frac{\partial}{\partial x_i}(\rho\tilde{\nu}u_i) = G_\nu + \frac{1}{\sigma_{\tilde{\nu}}}\left[\frac{\partial}{\partial x_j}\left\{(\mu+\rho\tilde{\nu})\frac{\partial\tilde{\nu}}{\partial x_j}\right\} + C_{b2}\rho\left(\frac{\partial\tilde{\nu}}{\partial x_j}\right)^2\right] - Y_\nu + S_{\tilde{\nu}} \tag{4.15}$$

where G_ν is the production of turbulent viscosity, and Y_ν is the destruction of turbulent viscosity that occurs in the near-wall region due to wall blocking and viscous damping. $\sigma_{\tilde{\nu}}$ and C_{b2} are the constants and ν is the molecular kinematic viscosity. $S_{\tilde{\nu}}$ is a user-defined source term. Note that since the turbulence kinetic energy, k, is not calculated in the Spalart-Allmaras model, the last term in Equation 4.14 (p. 44) is ignored when estimating the Reynolds stresses.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

4.2.3. Modeling the Turbulent Viscosity

The turbulent viscosity, μ_t, is computed from

$$\mu_t = \rho \tilde{\nu} f_{v1} \tag{4.16}$$

where the viscous damping function, f_{v1}, is given by

$$f_{v1} = \frac{\chi^3}{\chi^3 + C_{v1}^3} \tag{4.17}$$

and

$$\chi \equiv \frac{\tilde{\nu}}{\nu} \tag{4.18}$$

4.2.4. Modeling the Turbulent Production

The production term, G_ν, is modeled as

$$G_\nu = C_{b1} \rho \tilde{S} \tilde{\nu} \tag{4.19}$$

where

$$\tilde{S} \equiv S + \frac{\tilde{\nu}}{\kappa^2 d^2} f_{v2} \tag{4.20}$$

and

$$f_{v2} = 1 - \frac{\chi}{1 + \chi f_{v1}} \tag{4.21}$$

C_{b1} and κ are constants, d is the distance from the wall, and S is a scalar measure of the deformation tensor. By default in Ansys Fluent, as in the original model proposed by Spalart and Allmaras, S is based on the magnitude of the vorticity:

$$S \equiv \sqrt{2 \Omega_{ij} \Omega_{ij}} \tag{4.22}$$

where Ω_{ij} is the mean rate-of-rotation tensor and is defined by

$$\Omega_{ij} = \frac{1}{2} \left(\frac{\partial u_i}{\partial x_j} - \frac{\partial u_j}{\partial x_i} \right) \tag{4.23}$$

The justification for the default expression for S is that, for shear flows, vorticity and strain rate are identical. Vorticity has the advantage of being zero in inviscid flow regions like stagnation lines, where turbulence production due to strain rate can be unphysical. However, an alternative formulation has been proposed [129] (p. 1060) and incorporated into Ansys Fluent.

This modification combines the measures of both vorticity and the strain tensors in the definition of S:

$$S \equiv |\Omega_{ij}| + C_{prod} \min \left(0, |S_{ij}| - |\Omega_{ij}| \right) \tag{4.24}$$

where

$$C_{prod} = 2.0, \; |\Omega_{ij}| \equiv \sqrt{2 \Omega_{ij} \Omega_{ij}}, \; |S_{ij}| \equiv \sqrt{2 S_{ij} S_{ij}}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

with the mean strain rate, S_{ij}, defined as

$$S_{ij}=\frac{1}{2}\left(\frac{\partial u_j}{\partial x_i}+\frac{\partial u_i}{\partial x_j}\right) \tag{4.25}$$

Including both the rotation and strain tensors reduces the production of eddy viscosity and consequently reduces the eddy viscosity itself in regions where the measure of vorticity exceeds that of strain rate. One such example can be found in vortical flows, that is, flow near the core of a vortex subjected to a pure rotation where turbulence is known to be suppressed. Including both the rotation and strain tensors more accurately accounts for the effects of rotation on turbulence. The default option (including the rotation tensor only) tends to overpredict the production of eddy viscosity and hence over-predicts the eddy viscosity itself inside vortices.

You can select the modified form for calculating production in the Viscous Model Dialog Box.

4.2.5. Modeling the Turbulent Destruction

The destruction term is modeled as

$$Y_v=C_{w1}\rho f_w\left(\frac{\tilde{\nu}}{d}\right)^2 \tag{4.26}$$

where

$$f_w=g\left[\frac{1+C_{w3}^6}{g^6+C_{w3}^6}\right]^{1/6} \tag{4.27}$$

$$g=r+C_{w2}\left(r^6-r\right) \tag{4.28}$$

$$r\equiv\frac{\tilde{\nu}}{\tilde{S}\kappa^2 d^2} \tag{4.29}$$

C_{w1}, C_{w2}, and C_{w3} are constants, and $\tilde{S}$ is given by Equation 4.20 (p. 46). Note that the modification described above to include the effects of mean strain on S will also affect the value of $\tilde{S}$ used to compute r.

4.2.6. Model Constants

The model constants C_{b1}, C_{b2}, $\sigma_{\tilde{\nu}}$,C_{v1}, C_{w1}, C_{w2}, C_{w3}, and κ have the following default values [618] (p. 1089):

$$C_{b1}=0.1355,\ C_{b2}=0.622,\ \sigma_{\tilde{\nu}}=\frac{2}{3},C_{v1}=7.1$$

$$C_{w1}=\frac{C_{b1}}{\kappa^2}+\frac{\left(1+C_{b2}\right)}{\sigma_{\tilde{\nu}}},\ C_{w2}=0.3,\ C_{w3}=2.0,\ \kappa=0.41$$

4.2.7. Wall Boundary Conditions

The Spalart-Allmaras model has been extended within Ansys Fluent with a y^+-insensitive wall treatment, which automatically blends all solution variables from their viscous sublayer formulation

$$\frac{u}{u_\tau}=\frac{\rho u_\tau y}{\mu} \tag{4.30}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

to the corresponding logarithmic layer values depending on y^+.

$$\frac{u}{u_\tau}=\frac{1}{\kappa}\ln E\left(\frac{\rho u_\tau y}{\mu}\right) \tag{4.31}$$

where u is the velocity parallel to the wall, u_τ is the friction velocity, y is the distance from the wall, κ is the von Kármán constant (0.4187), and $E=9.793$.

The blending is calibrated to also cover intermediate y^+ values in the buffer layer $(1<y^+<30)$.

4.2.7.1. Treatment of the Spalart-Allmaras Model for Icing Simulations

On the basis of the standard Spalart-Allmaras model equation, the Boeing Extension [33] (p. 1054) has been adopted to account for wall roughness. In this model, the non-zero wall value of the transported variable (directly solved from the S-A equation), $\tilde{\nu}_w$, is estimated to mimic roughness effects by replacing the wall condition $\tilde{\nu}_w=0$ with:

$$\frac{\partial\tilde{\nu}}{\partial n}=\frac{\tilde{\nu}_w}{d_w+d_o(h_s)} \tag{4.32}$$

where n is the wall normal, d_w is the minimum cell-to-face distance between the wall and the first cell near the wall, and $d_o(h_s)$ is a length introduced to impose an offset, depending on the local roughness height, h_s:

$$d_o(h_s)=e^{(-8.5x)}d_o(h_s)\approx 0.03h_s \tag{4.33}$$

Then the wall turbulent kinematic viscosity, $\nu_{t,w}$, is obtained as follows:

$$\nu_{t,w}=f_{v1}\tilde{\nu}_w=\mu_\tau\kappa[d_w+d_o(h_s)] \tag{4.34}$$

where f_{v1} is the standard model function in the Spalart-Allmaras model. As the roughness effect is strong, the turbulent viscosity should be large compared to the laminar viscosity at the wall, then $f_{v1}\rightarrow 1$, and therefore:

$$\nu_{t,w}=\tilde{\nu}_w \tag{4.35}$$

Also in Equation 4.34 (p. 48), κ is the Von Karman constant, and u_τ is the wall friction velocity:

$$u_\tau^2=(1+\nu_{t,w})\frac{\partial u}{\partial n}=[1+u_\tau\kappa(d_w+d_o(h_s))]\frac{\partial u}{\partial n} \tag{4.36}$$

Furthermore, to achieve good predictions for smaller roughness, Aupoix and Spalart [33] (p. 1054) proposed that the function f_{v1} should be altered by modifying the quantity χ in the Spalart-Allmaras model equation:

$$\chi=\frac{\tilde{\nu}_t}{\nu}+0.5\frac{h_s}{d_w+d_o(h_s)} \tag{4.37}$$

4.2.8. Convective Heat and Mass Transfer Modeling

In Ansys Fluent, turbulent heat transport is modeled using the concept of the Reynolds analogy to turbulent momentum transfer. The "modeled" energy equation is as follows:

$$\frac{\partial}{\partial t}(\rho E)+\frac{\partial}{\partial x_i}\left[u_i(\rho E+p)\right]=\frac{\partial}{\partial x_j}\left[\left(k+\frac{c_p\mu_t}{Pr_t}\right)\frac{\partial T}{\partial x_j}+u_i(\tau_{ij})_{eff}\right]+S_h \tag{4.38}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

where k, in this case, is the thermal conductivity, E is the total energy, and $\left(\tau_{ij}\right)_{eff}$ is the deviatoric stress tensor, defined as

$$\left(\tau_{ij}\right)_{eff}=\mu_{eff}\left(\frac{\partial u_j}{\partial x_i}+\frac{\partial u_i}{\partial x_j}\right)-\frac{2}{3}\mu_{eff}\frac{\partial u_k}{\partial x_k}\delta_{ij}$$

4.3. Standard, RNG, and Realizable k-ε Models

This section describes the theory behind the Standard, RNG, and Realizable k-ε models. Information is presented in the following sections:

4.3.1. Standard k-ε Model

4.3.2. RNG k-ε Model

4.3.3. Realizable k-ε Model

4.3.4. Modeling Turbulent Production in the k-ε Models

4.3.5. Remarks on the Linearization of Destruction Terms

4.3.6. Effects of Buoyancy on Turbulence in the k-ε Models

4.3.7. Turbulence Damping

4.3.8. Effects of Compressibility on Turbulence in the k-ε Models

4.3.9. Convective Heat and Mass Transfer Modeling in the k-ε Models

For details about using the models in Ansys Fluent, see Modeling Turbulence and Setting Up the k-ε Model in the User's Guide.

This section presents the standard, RNG, and realizable k-ε models. All three models have similar forms, with transport equations for k and ε. The major differences in the models are as follows:

- the method of calculating turbulent viscosity
- the turbulent Prandtl numbers governing the turbulent diffusion of k and ε
- the generation and destruction terms in the ε equation

The transport equations, the methods of calculating turbulent viscosity, and model constants are presented separately for each model. The features that are essentially common to all models follow, including turbulent generation due to shear buoyancy, accounting for the effects of compressibility, and modeling heat and mass transfer.

4.3.1. Standard k-ε Model

4.3.1.1. Overview

Two-equation turbulence models allow the determination of both, a turbulent length and time scale by solving two separate transport equations. The standard k-ε model in Ansys Fluent falls within this class of models and has become the workhorse of practical engineering flow calculations in the time since it was proposed by Launder and Spalding [340] (p. 1072). Robustness, economy, and reasonable accuracy for a wide range of turbulent flows explain its popularity in industrial flow

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

and heat transfer simulations. It is a semi-empirical model, and the derivation of the model equations relies on phenomenological considerations and empiricism.

The standard k-ε model [340] (p. 1072) is a model based on model transport equations for the turbulence kinetic energy (k) and its dissipation rate (ε). The model transport equation for k is derived from the exact equation, while the model transport equation for ε was obtained using physical reasoning and bears little resemblance to its mathematically exact counterpart.

In the derivation of the k-ε model, the assumption is that the flow is fully turbulent, and the effects of molecular viscosity are negligible. The standard k-ε model is therefore valid only for fully turbulent flows.

As the strengths and weaknesses of the standard k-ε model have become known, modifications have been introduced to improve its performance. Two of these variants are available in Ansys Fluent: the RNG k-ε model [718] (p. 1094) and the realizable k-ε model [588] (p. 1087).

4.3.1.2. Transport Equations for the Standard k-ε Model

The turbulence kinetic energy, k, and its rate of dissipation, ε, are obtained from the following transport equations:

$$\frac{\partial}{\partial t}(\rho k)+\frac{\partial}{\partial x_i}(\rho k u_i)=\frac{\partial}{\partial x_j}\left[\left(\mu+\frac{\mu_t}{\sigma_k}\right)\frac{\partial k}{\partial x_j}\right]+G_k+G_b-\rho\varepsilon-Y_M+S_k \tag{4.39}$$

and

$$\frac{\partial}{\partial t}(\rho \varepsilon)+\frac{\partial}{\partial x_i}(\rho \varepsilon u_i)=\frac{\partial}{\partial x_j}\left[\left(\mu+\frac{\mu_t}{\sigma_\varepsilon}\right)\frac{\partial \varepsilon}{\partial x_j}\right]+C_{1\varepsilon}\frac{\varepsilon}{k}(G_k+C_{3\varepsilon}G_b)-C_{2\varepsilon}\rho\frac{\varepsilon^2}{k}+S_\varepsilon \tag{4.40}$$

In these equations, G_k represents the generation of turbulence kinetic energy due to the mean velocity gradients, calculated as described in Modeling Turbulent Production in the k-ε Models (p. 57). G_b is the generation of turbulence kinetic energy due to buoyancy, calculated as described in Effects of Buoyancy on Turbulence in the k-ε Models (p. 58). Y_M represents the contribution of the fluctuating dilatation in compressible turbulence to the overall dissipation rate, calculated as described in Effects of Compressibility on Turbulence in the k-ε Models (p. 59). $C_{1\varepsilon}$, $C_{2\varepsilon}$, and $C_{3\varepsilon}$ are constants. σ_k and σ_ε are the turbulent Prandtl numbers for k and ε, respectively. S_k and S_ε are user-defined source terms.

4.3.1.3. Modeling the Turbulent Viscosity

The turbulent (or eddy) viscosity, μ_t, is computed by combining k and ε as follows:

$$\mu_t=\rho C_\mu\frac{k^2}{\varepsilon} \tag{4.41}$$

where C_μ is a constant.

4.3.1.4. Model Constants

The model constants $C_{1\varepsilon}$, $C_{2\varepsilon}$, C_μ, σ_k, and σ_ε have the following default values [340] (p. 1072):

$$C_{1\varepsilon}=1.44,\ C_{2\varepsilon}=1.92,\ C_\mu=0.09,\ \sigma_k=1.0,\ \sigma_\varepsilon=1.3$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

These default values have been determined from experiments for fundamental turbulent flows including frequently encountered shear flows like boundary layers, mixing layers and jets as well as for decaying isotropic grid turbulence. They have been found to work fairly well for a wide range of wall-bounded and free shear flows.

Although the default values of the model constants are the standard ones most widely accepted, you can change them (if needed) in the Viscous Model Dialog Box.

4.3.2. RNG k-ε Model

4.3.2.1. Overview

The RNG k-ε model was derived using a statistical technique called renormalization group theory. It is similar in form to the standard k-ε model, but includes the following refinements:

- The RNG model has an additional term in its ε equation that improves the accuracy for rapidly strained flows.
- The effect of swirl on turbulence is included in the RNG model, enhancing accuracy for swirling flows.
- The RNG theory provides an analytical formula for turbulent Prandtl numbers, while the standard k-ε model uses user-specified, constant values.
- While the standard k-ε model is a high-Reynolds number model, the RNG theory provides an analytically-derived differential formula for effective viscosity that accounts for low-Reynolds number effects. Effective use of this feature does, however, depend on an appropriate treatment of the near-wall region.

These features make the RNG k-ε model more accurate and reliable for a wider class of flows than the standard k-ε model.

The RNG-based k-ε turbulence model is derived from the instantaneous Navier-Stokes equations, using a mathematical technique called "renormalization group" (RNG) methods. The analytical derivation results in a model with constants different from those in the standard k-ε model, and additional terms and functions in the transport equations for k and ε. A more comprehensive description of RNG theory and its application to turbulence can be found in [496] (p. 1082).

4.3.2.2. Transport Equations for the RNG k-ε Model

The RNG k-ε model has a similar form to the standard k-ε model:

$$\frac{\partial}{\partial t}(\rho k)+\frac{\partial}{\partial x_i}(\rho k u_i)=\frac{\partial}{\partial x_j}\left(\alpha_k \mu_{eff}\frac{\partial k}{\partial x_j}\right)+G_k+G_b-\rho\varepsilon-Y_M+S_k \tag{4.42}$$

and

$$\frac{\partial}{\partial t}(\rho \varepsilon)+\frac{\partial}{\partial x_i}(\rho \varepsilon u_i)=\frac{\partial}{\partial x_j}\left(\alpha_\varepsilon \mu_{eff}\frac{\partial \varepsilon}{\partial x_j}\right)+C_{1\varepsilon}\frac{\varepsilon}{k}(G_k+C_{3\varepsilon}G_b)-C_{2\varepsilon}\rho\frac{\varepsilon^2}{k}-R_\varepsilon+S_\varepsilon \tag{4.43}$$

In these equations, G_k represents the generation of turbulence kinetic energy due to the mean velocity gradients, calculated as described in Modeling Turbulent Production in the k-ε Models (p. 57). G_b is the generation of turbulence kinetic energy due to buoyancy, calculated as described in Effects

of Buoyancy on Turbulence in the k-ε Models (p. 58). Y_M represents the contribution of the fluctuating dilatation in compressible turbulence to the overall dissipation rate, calculated as described in Effects of Compressibility on Turbulence in the k-ε Models (p. 59). The quantities α_k and α_ε are the inverse effective Prandtl numbers for k and ε, respectively. S_k and S_ε are user-defined source terms.

4.3.2.3. Modeling the Effective Viscosity

The scale elimination procedure in RNG theory results in a differential equation for turbulent viscosity:

$$d\left(\frac{\rho^2 k}{\sqrt{\varepsilon\mu}}\right) = 1.72\frac{\hat{\nu}}{\sqrt{\hat{\nu}^3 - 1 + C_\nu}}d\hat{\nu} \tag{4.44}$$

where

$$\hat{\nu} = \frac{\mu_{eff}}{\mu}$$

$$C_\nu \approx 100$$

Equation 4.44 (p. 52) is integrated to obtain an accurate description of how the effective turbulent transport varies with the effective Reynolds number (or eddy scale), allowing the model to better handle low-Reynolds number and near-wall flows.

In the high-Reynolds number limit, Equation 4.44 (p. 52) gives

$$\mu_t = \rho C_\mu \frac{k^2}{\varepsilon} \tag{4.45}$$

with $C_\mu = 0.0845$, derived using RNG theory. It is interesting to note that this value of C_μ is very close to the empirically-determined value of 0.09 used in the standard k-ε model.

In Ansys Fluent, by default, the effective viscosity is computed using the high-Reynolds number form in Equation 4.45 (p. 52). However, there is an option available that allows you to use the differential relation given in Equation 4.44 (p. 52) when you need to include low-Reynolds number effects.

4.3.2.4. RNG Swirl Modification

Turbulence, in general, is affected by rotation or swirl in the mean flow. The RNG model in Ansys Fluent provides an option to account for the effects of swirl or rotation by modifying the turbulent viscosity appropriately. The modification takes the following functional form:

$$\mu_t = \mu_{t0} f\left(\alpha_s, \Omega, \frac{k}{\varepsilon}\right) \tag{4.46}$$

where μ_{t0} is the value of turbulent viscosity calculated without the swirl modification using either Equation 4.44 (p. 52) or Equation 4.45 (p. 52). Ω is a characteristic swirl number evaluated within Ansys Fluent, and α_s is a swirl constant that assumes different values depending on whether the flow is swirl-dominated or only mildly swirling. This swirl modification always takes effect for axisymmetric, swirling flows and three-dimensional flows when the RNG model is selected. For mildly swirling flows (the default in Ansys Fluent), α_s is set to 0.07. For strongly swirling flows, however, a higher value of α_s can be used.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

4.3.2.5. Calculating the Inverse Effective Prandtl Numbers

The inverse effective Prandtl numbers, α_k and α_ε, are computed using the following formula derived analytically by the RNG theory:

$$\left|\frac{\alpha-1.3929}{\alpha_0-1.3929}\right|^{0.6321}\left|\frac{\alpha+2.3929}{\alpha_0+2.3929}\right|^{0.3679}=\frac{\mu_{mol}}{\mu_{eff}} \tag{4.47}$$

where $\alpha_0=1.0$. In the high-Reynolds number limit ($\frac{\mu_{mol}}{\mu_{eff}}\ll 1$), $\alpha_k=\alpha_\varepsilon\approx 1.393$.

4.3.2.6. The R-ε Term in the ε Equation

The main difference between the RNG and standard k-ε models lies in the additional term in the ε equation given by

$$R_\varepsilon=\frac{C_\mu\rho\eta^3\left(1-\eta/\eta_0\right)}{1+\beta\eta^3}\frac{\varepsilon^2}{k} \tag{4.48}$$

where $\eta\equiv Sk/\varepsilon$, $\eta_0=4.38$, $\beta=0.012$.

The effects of this term in the RNG ε equation can be seen more clearly by rearranging Equation 4.43 (p. 51). Using Equation 4.48 (p. 53), the third and fourth terms on the right-hand side of Equation 4.43 (p. 51) can be merged, and the resulting ε equation can be rewritten as

$$\frac{\partial}{\partial t}\left(\rho\varepsilon\right)+\frac{\partial}{\partial x_i}\left(\rho\varepsilon u_i\right)=\frac{\partial}{\partial x_j}\left(\alpha_\varepsilon\mu_{eff}\frac{\partial\varepsilon}{\partial x_j}\right)+C_{1\varepsilon}\frac{\varepsilon}{k}\left(G_k+C_{3\varepsilon}G_b\right)-C_{2\varepsilon}^*\rho\frac{\varepsilon^2}{k} \tag{4.49}$$

where $C_{2\varepsilon}^*$ is given by

$$C_{2\varepsilon}^*\equiv C_{2\varepsilon}+\frac{C_\mu\eta^3\left(1-\eta/\eta_0\right)}{1+\beta\eta^3} \tag{4.50}$$

In regions where $\eta<\eta_0$, the R term makes a positive contribution, and $C_{2\varepsilon}^*$ becomes larger than $C_{2\varepsilon}$. In the logarithmic layer, for instance, it can be shown that $\eta\approx 3.0$, giving $C_{2\varepsilon}^*\approx 2.0$, which is close in magnitude to the value of $C_{2\varepsilon}$ in the standard k-ε model (1.92). As a result, for weakly to moderately strained flows, the RNG model tends to give results largely comparable to the standard k-ε model.

In regions of large strain rate ($\eta>\eta_0$), however, the R term makes a negative contribution, making the value of $C_{2\varepsilon}^*$ less than $C_{2\varepsilon}$. In comparison with the standard k-ε model, the smaller destruction of ε augments ε, reducing k and, eventually, the effective viscosity. As a result, in rapidly strained flows, the RNG model yields a lower turbulent viscosity than the standard k-ε model.

Thus, the RNG model is more responsive to the effects of rapid strain and streamline curvature than the standard k-ε model, which explains the superior performance of the RNG model for certain classes of flows.

4.3.2.7. Model Constants

The model constants $C_{1\varepsilon}$ and $C_{2\varepsilon}$ in Equation 4.43 (p. 51) have values derived analytically by the RNG theory. These values, used by default in Ansys Fluent, are

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$C_{1\varepsilon}=1.42,\ C_{2\varepsilon}=1.68$$

4.3.3. Realizable k-ε Model

4.3.3.1. Overview

The realizable k-ε model [588] (p. 1087) differs from the standard k-ε model in two important ways:

- The realizable k-ε model contains an alternative formulation for the turbulent viscosity.
- A modified transport equation for the dissipation rate, ε, has been derived from an exact equation for the transport of the mean-square vorticity fluctuation.

The term "realizable" means that the model satisfies certain mathematical constraints on the Reynolds stresses, consistent with the physics of turbulent flows. Neither the standard k-ε model nor the RNG k-ε model is realizable.

To understand the mathematics behind the realizable k-ε model, consider combining the Boussinesq relationship (Equation 4.14 (p. 44)) and the eddy viscosity definition (Equation 4.41 (p. 50)) to obtain the following expression for the normal Reynolds stress in an incompressible strained mean flow:

$$\overline{u^2}=\frac{2}{3}k-2\nu_t\frac{\partial U}{\partial x} \tag{4.51}$$

Using Equation 4.41 (p. 50) for $\nu_t \equiv \mu_t / \rho$, one obtains the result that the normal stress, $\overline{u^2}$, which by definition is a positive quantity, becomes negative, that is, "non-realizable", when the strain is large enough to satisfy

$$\frac{k}{\varepsilon}\frac{\partial U}{\partial x}>\frac{1}{3C_\mu}\approx 3.7 \tag{4.52}$$

Similarly, it can also be shown that the Schwarz inequality for shear stresses ($\overline{u_\alpha}\overline{u_\beta}^2 \le \overline{u_\alpha^2 u_\beta^2}$; no summation over α and β) can be violated when the mean strain rate is large. The most straightforward way to ensure the realizability (positivity of normal stresses and Schwarz inequality for shear stresses) is to make C_μ variable by sensitizing it to the mean flow (mean deformation) and the turbulence (k, ε). The notion of variable C_μ is suggested by many modelers including Reynolds [549] (p. 1085), and is well substantiated by experimental evidence. For example, C_μ is found to be around 0.09 in the logarithmic layer of equilibrium boundary layers, and 0.05 in a strong homogeneous shear flow.

Both the realizable and RNG k-ε models have shown substantial improvements over the standard k-ε model where the flow features include strong streamline curvature, vortices, and rotation. Since the model is still relatively new, it is not clear in exactly which instances the realizable k-ε model consistently outperforms the RNG model. However, initial studies have shown that the realizable model provides the best performance of all the k-ε model versions for several validations of separated flows and flows with complex secondary flow features.

One of the weaknesses of the standard k-ε model or other traditional k-ε models lies with the modeled equation for the dissipation rate (ε). The well-known round-jet anomaly (named based on the finding that the spreading rate in planar jets is predicted reasonably well, but prediction of the spreading rate for axisymmetric jets is unexpectedly poor) is considered to be mainly due to the modeled dissipation equation.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

The realizable k-ε model proposed by Shih et al. [588] (p. 1087) was intended to address these deficiencies of traditional k-ε models by adopting the following:

- A new eddy-viscosity formula involving a variable C_μ originally proposed by Reynolds [549] (p. 1085).
- A new model equation for dissipation (ε) based on the dynamic equation of the mean-square vorticity fluctuation.

One limitation of the realizable k-ε model is that it produces non-physical turbulent viscosities in situations when the computational domain contains both rotating and stationary fluid zones (for example, multiple reference frames, rotating sliding meshes). This is due to the fact that the realizable k-ε model includes the effects of mean rotation in the definition of the turbulent viscosity (see Equation 4.55 (p. 56) – Equation 4.57 (p. 56)). This extra rotation effect has been tested on single moving reference frame systems and showed superior behavior over the standard k-ε model. However, due to the nature of this modification, its application to multiple reference frame systems should be taken with some caution. See Modeling the Turbulent Viscosity (p. 56) for information about how to include or exclude this term from the model.

4.3.3.2. Transport Equations for the Realizable k-ε Model

The modeled transport equations for k and ε in the realizable k-ε model are

$$\frac{\partial}{\partial t}(\rho k)+\frac{\partial}{\partial x_j}(\rho k u_j)=\frac{\partial}{\partial x_j}\left[\left(\mu+\frac{\mu_t}{\sigma_k}\right)\frac{\partial k}{\partial x_j}\right]+G_k+G_b-\rho\varepsilon-Y_M+S_k \tag{4.53}$$

and

$$\begin{aligned}\frac{\partial}{\partial t}(\rho\varepsilon)+\frac{\partial}{\partial x_j}(\rho\varepsilon u_j) &= \frac{\partial}{\partial x_j}\left[\left(\mu+\frac{\mu_t}{\sigma_\varepsilon}\right)\frac{\partial \varepsilon}{\partial x_j}\right] \\ &\quad +\rho C_1 S\varepsilon-\rho C_2\frac{\varepsilon^2}{k+\sqrt{\nu\varepsilon}}+C_{1\varepsilon}\frac{\varepsilon}{k}C_{3\varepsilon}G_b+S_\varepsilon\end{aligned} \tag{4.54}$$

where

$$C_1=\max\left[0.43,\frac{\eta}{\eta+5}\right],\ \eta=S\frac{k}{\varepsilon},\ S=\sqrt{2S_{ij}S_{ij}}$$

In these equations, G_k represents the generation of turbulence kinetic energy due to the mean velocity gradients, calculated as described in Modeling Turbulent Production in the k-ε Models (p. 57). G_b is the generation of turbulence kinetic energy due to buoyancy, calculated as described in Effects of Buoyancy on Turbulence in the k-ε Models (p. 58). Y_M represents the contribution of the fluctuating dilatation in compressible turbulence to the overall dissipation rate, calculated as described in Effects of Compressibility on Turbulence in the k-ε Models (p. 59). C_2 and $C_{1\varepsilon}$ are constants. σ_k and σ_ε are the turbulent Prandtl numbers for k and ε, respectively. S_k and S_ε are user-defined source terms.

Note that the k equation (Equation 4.53 (p. 55)) is the same as that in the standard k-ε model (Equation 4.39 (p. 50)) and the RNG k-ε model (Equation 4.42 (p. 51)), except for the model constants. However, the form of the ε equation is quite different from those in the standard and RNG-based k-ε models (Equation 4.40 (p. 50) and Equation 4.43 (p. 51)). One of the noteworthy features is that the production term in the ε equation (the second term on the right-hand side of Equation 4.54 (p. 55)) does not involve the production of k; that is, it does not contain the same G_k

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

term as the other k-ε models. It is believed that the present form better represents the spectral energy transfer. Another desirable feature is that the destruction term (the third term on the right-hand side of Equation 4.54 (p. 55)) does not have any singularity; that is, its denominator never vanishes, even if k vanishes or becomes smaller than zero. This feature is contrasted with traditional k-ε models, which have a singularity due to k in the denominator.

This model has been extensively validated for a wide range of flows [307] (p. 1071), [588] (p. 1087), including rotating homogeneous shear flows, free flows including jets and mixing layers, channel and boundary layer flows, and separated flows. For all these cases, the performance of the model has been found to be substantially better than that of the standard k-ε model. Especially noteworthy is the fact that the realizable k-ε model resolves the round-jet anomaly; that is, it predicts the spreading rate for axisymmetric jets as well as that for planar jets.

4.3.3.3. Modeling the Turbulent Viscosity

As in other k-ε models, the eddy viscosity is computed from

$$\mu_t = \rho C_\mu \frac{k^2}{\varepsilon} \tag{4.55}$$

The difference between the realizable k-ε model and the standard and RNG k-ε models is that C_μ is no longer constant. It is computed from

$$C_\mu = \frac{1}{A_0 + A_s \frac{kU^*}{\varepsilon}} \tag{4.56}$$

where

$$U^* \equiv \sqrt{S_{ij}S_{ij} + \tilde{\Omega}_{ij}\tilde{\Omega}_{ij}} \tag{4.57}$$

and

$$\begin{aligned} \tilde{\Omega}_{ij} &= \Omega_{ij} - 2\varepsilon_{ijk}\omega_k \\ \Omega_{ij} &= \overline{\Omega_{ij}} - \varepsilon_{ijk}\omega_k \end{aligned}$$

where $\overline{\Omega_{ij}}$ is the mean rate-of-rotation tensor viewed in a moving reference frame with the angular velocity ω_k. The model constants A_0 and A_s are given by

$$A_0 = 4.04, \; A_s = \sqrt{6}\cos\varphi \tag{4.58}$$

where

$$\varphi = \frac{1}{3}\cos^{-1}\left(\sqrt{6}\,W\right), \; W = \frac{S_{ij}S_{jk}S_{ki}}{\tilde{S}^3}, \; \tilde{S} = \sqrt{S_{ij}S_{ij}}, \; S_{ij} = \frac{1}{2}\left(\frac{\partial u_j}{\partial x_i} + \frac{\partial u_i}{\partial x_j}\right) \tag{4.59}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

It can be seen that C_μ is a function of the mean strain and rotation rates, the angular velocity of the system rotation, and the turbulence fields (k and ε). C_μ in Equation 4.55 (p. 56) can be shown to recover the standard value of 0.09 for an inertial sublayer in an equilibrium boundary layer.

Important:

In Ansys Fluent, the term $-2\varepsilon_{ijk}\omega_k$ is, by default, not included in the calculation of $\tilde{\Omega}_{ij}$. This is an extra rotation term that is not compatible with cases involving sliding meshes or multiple reference frames. If you want to include this term in the model, you can enable it by using the `define/models/viscous/turbulence-expert/rke-cmu-rotation-term?` text command and entering `yes` at the prompt.

4.3.3.4. Model Constants

The model constants C_2, σ_k, and σ_ε have been established to ensure that the model performs well for certain canonical flows. The model constants are

$$C_{1\varepsilon}=1.44,\ C_2=1.9,\ \sigma_k=1.0,\ \sigma_\varepsilon=1.2$$

4.3.4. Modeling Turbulent Production in the k-ε Models

The term G_k, representing the production of turbulence kinetic energy, is modeled identically for the standard, RNG, and realizable k-ε models. From the exact equation for the transport of k, this term may be defined as

$$G_k=-\rho\overline{u_i'u_j'}\frac{\partial u_j}{\partial x_i} \tag{4.60}$$

To evaluate G_k in a manner consistent with the Boussinesq hypothesis,

$$G_k=\mu_t S^2 \tag{4.61}$$

where S is the modulus of the mean rate-of-strain tensor, defined as

$$S\equiv\sqrt{2S_{ij}S_{ij}} \tag{4.62}$$

Important:

When using the high-Reynolds number k-ε versions, the effective viscosity $\left(\mu_t+C\cdot\mu\right)$ is used in lieu of μ_t in Equation 4.61 (p. 57). The coefficient C has a default value of 1 and can be accessed via the `rpvar 'coeff-ke-mu-prod`.

4.3.5. Remarks on the Linearization of Destruction Terms

Linearizing the destruction terms of the transport equations for k and ε the ratio ε/k appears. It is reformulated to improve robustness when using high-Reynolds number versions of the Standard-k-ε and RNG model. The formulation reads:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$\frac{\varepsilon}{k} \rightarrow \frac{\rho C_\mu k}{\mu_t + C \cdot \mu} \tag{4.63}$$

The coefficient C has a default value of 1 and can be accesses via the `rpvar 'coeff-ke-mu-dok`.

Note:

if an UDF is used for the eddy viscosity, then this relation is not applied and instead the ratio ε/k is evaluated directly.

4.3.6. Effects of Buoyancy on Turbulence in the k-ε Models

When a nonzero gravity field and temperature gradient are present simultaneously, the k-ε models in Ansys Fluent can account for the generation of k due to buoyancy (G_b in Equation 4.39 (p. 50), Equation 4.42 (p. 51), and Equation 4.53 (p. 55)), and the corresponding contribution to the production of ε in Equation 4.40 (p. 50), Equation 4.43 (p. 51), and Equation 4.54 (p. 55).

The generation of turbulence due to buoyancy is given by

$$G_b = \beta g_i \frac{\mu_t}{Pr_t} \frac{\partial T}{\partial x_i} \tag{4.64}$$

where Pr_t is the turbulent Prandtl number for energy and g_i is the component of the gravitational vector in the i^{th} direction. For the standard and realizable k-ε models, the default value of Pr_t is 0.85. For non-premixed and partially premixed combustion models, Pr_t is set equal to the PDF Schmidt number to ensure a Lewis number equal to unity. In the case of the RNG k-ε model, $Pr_t = 1/\alpha$, where α is given by Equation 4.47 (p. 53), but with $\alpha_0 = 1/Pr = k/\mu c_p$. The coefficient of thermal expansion, β, is defined as

$$\beta = -\frac{1}{\rho}\left(\frac{\partial \rho}{\partial T}\right)_p \tag{4.65}$$

For ideal gases, Equation 4.64 (p. 58) reduces to

$$G_b = -g_i \frac{\mu_t}{\rho Pr_t} \frac{\partial \rho}{\partial x_i} \tag{4.66}$$

It can be seen from the transport equations for k (Equation 4.39 (p. 50), Equation 4.42 (p. 51), and Equation 4.53 (p. 55)) that turbulence kinetic energy tends to be augmented ($G_b > 0$) in unstable stratification. For stable stratification, buoyancy tends to suppress the turbulence ($G_b < 0$). In Ansys Fluent, the effects of buoyancy on the generation of k are included by default when you have both a nonzero gravity field and a nonzero temperature (or density) gradient.

While the buoyancy effects on the generation of k are relatively well understood, the effect on ε is less clear. In Ansys Fluent, by default, the buoyancy effects on ε are neglected simply by setting G_b to zero in the transport equation for ε (Equation 4.40 (p. 50), Equation 4.43 (p. 51), or Equation 4.54 (p. 55)).

However, you can include the buoyancy effects on ε in the Viscous Model Dialog Box. In this case, the value of G_b given by Equation 4.66 (p. 58) is used in the transport equation for ε (Equation 4.40 (p. 50), Equation 4.43 (p. 51), or Equation 4.54 (p. 55)).

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

The degree to which ε is affected by the buoyancy is determined by the constant $C_{3\varepsilon}$. In Ansys Fluent, $C_{3\varepsilon}$ is not specified, but is instead calculated according to the following relation [237] (p. 1066):

$$C_{3\varepsilon}=tanh\left|\frac{v}{u}\right| \tag{4.67}$$

where v is the component of the flow velocity parallel to the gravitational vector and u is the component of the flow velocity perpendicular to the gravitational vector. In this way, $C_{3\varepsilon}$ will become 1 for buoyant shear layers for which the main flow direction is aligned with the direction of gravity. For buoyant shear layers that are perpendicular to the gravitational vector, $C_{3\varepsilon}$ will become zero.

4.3.7. Turbulence Damping

For details about turbulence damping, see Turbulence Damping (p. 167).

4.3.8. Effects of Compressibility on Turbulence in the k-ε Models

For high-Mach-number flows, compressibility affects turbulence through so-called "dilatation dissipation", which is normally neglected in the modeling of incompressible flows [705] (p. 1094). Neglecting the dilatation dissipation fails to predict the observed decrease in spreading rate with increasing Mach number for compressible mixing and other free shear layers. To account for these effects in the k-ε models in Ansys Fluent, the dilatation dissipation term, Y_M, can be included in the k equation. This term is modeled according to a proposal by Sarkar [563] (p. 1085):

$$Y_M=2\rho\varepsilon M_t^2 \tag{4.68}$$

where M_t is the turbulent Mach number, defined as

$$M_t=\sqrt{\frac{k}{a^2}} \tag{4.69}$$

where a ($\equiv\sqrt{\gamma RT}$) is the speed of sound.

Note:

The Sarkar model has been tested for a very limited number of free shear test cases, and should be used with caution (and only when truly necessary), as it can negatively affect the wall boundary layer even at transonic and supersonic Mach numbers. It is disabled by default. For details, see Model Enhancements in the *Fluent User's Guide*.

4.3.9. Convective Heat and Mass Transfer Modeling in the k-ε Models

In Ansys Fluent, turbulent heat transport is modeled using the concept of Reynolds' analogy to turbulent momentum transfer. The "modeled" energy equation is therefore given by

$$\frac{\partial}{\partial t}\left(\rho E\right)+\frac{\partial}{\partial x_i}\left[u_i\left(\rho E+p\right)\right]=\frac{\partial}{\partial x_j}\left(k_{eff}\frac{\partial T}{\partial x_j}+u_i\left(\tau_{ij}\right)_{eff}\right)+S_h \tag{4.70}$$

where E is the total energy, k_{eff} is the effective thermal conductivity, and

$\left(\tau_{ij}\right)_{eff}$ is the deviatoric stress tensor, defined as

$$\left(\tau_{ij}\right)_{eff}=\mu_{eff}\left(\frac{\partial u_j}{\partial x_i}+\frac{\partial u_i}{\partial x_j}\right)-\frac{2}{3}\mu_{eff}\frac{\partial u_k}{\partial x_k}\delta_{ij}$$

The term involving $\left(\tau_{ij}\right)_{eff}$ represents the viscous heating, and is always computed in the density-based solvers. It is not computed by default in the pressure-based solver, but it can be enabled in the Viscous Model Dialog Box.

Additional terms may appear in the energy equation, depending on the physical models you are using. See Heat Transfer Theory (p. 172) for more details.

For the standard and realizable k-ε models, the effective thermal conductivity is given by

$$k_{eff}=k+\frac{c_p\mu_t}{Pr_t}$$

where k, in this case, is the thermal conductivity. The default value of the turbulent Prandtl number is 0.85. You can change the value of the turbulent Prandtl number in the Viscous Model Dialog Box.

For the RNG k-ε model, the effective thermal conductivity is

$$k_{eff}=\alpha c_p\mu_{eff}$$

where α is calculated from Equation 4.47 (p. 53), but with $\alpha_0=1/\ Pr=k/\ \mu c_p$.

The fact that α varies with $\mu_{mol}/\ \mu_{eff}$, as in Equation 4.47 (p. 53), is an advantage of the RNG k-ε model. It is consistent with experimental evidence indicating that the turbulent Prandtl number varies with the molecular Prandtl number and turbulence [295] (p. 1070). Equation 4.47 (p. 53) works well across a very broad range of molecular Prandtl numbers, from liquid metals ($Pr\approx10^{-2}$) to paraffin oils ($Pr\approx10^{3}$), which allows heat transfer to be calculated in low-Reynolds number regions. Equation 4.47 (p. 53) smoothly predicts the variation of effective Prandtl number from the molecular value ($\alpha=1/\ Pr$) in the viscosity-dominated region to the fully turbulent value ($\alpha=1.393$) in the fully turbulent regions of the flow.

Turbulent mass transfer is treated similarly. For the standard and realizable k-ε models, the default turbulent Schmidt number is 0.7. This default value can be changed in the Viscous Model Dialog Box. For the RNG model, the effective turbulent diffusivity for mass transfer is calculated in a manner that is analogous to the method used for the heat transport. The value of α_0 in Equation 4.47 (p. 53) is $\alpha_0=1/\ Sc$, where Sc is the molecular Schmidt number.

4.4. Standard, BSL, and SST k-ω Models

This section describes the theory behind the Standard, BSL, and SST k-ω models. Information is presented in the following sections:

4.4.1. Standard k-ω Model

4.4.2. Baseline (BSL) k-ω Model

4.4.3. Shear-Stress Transport (SST) k-ω Model

4.4.4. Effects of Buoyancy on Turbulence in the k-ω Models

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

4.4.5. Turbulence Damping

For details about using the models in Ansys Fluent, see Modeling Turbulence and Setting Up the k-ω Model in the User's Guide.

This section presents the standard [705] (p. 1094), baseline (BSL) [425] (p. 1077), and shear-stress transport (SST) [425] (p. 1077) k-ω models. All three models have similar forms, with transport equations for k and ω. The major ways in which the BSL and SST models [429] (p. 1078) differ from the standard model are as follows:

- gradual change from the standard k-ω model in the inner region of the boundary layer to a high-Reynolds number version of the k-ε model in the outer part of the boundary layer (BSL, SST)
- modified turbulent viscosity formulation to account for the transport effects of the principal turbulent shear stress (SST only)

The transport equations, methods of calculating turbulent viscosity, and methods of calculating model constants and other terms are presented separately for each model.

Low Reynolds number modifications have been proposed by Wilcox for the k-ω model and are available in Ansys Fluent. It is important to note that all k-ω models can be integrated through the viscous sub-layer without these terms. The terms were mainly added to reproduce the peak in the turbulence kinetic energy observed in DNS data very close to the wall. In addition, these terms affect the laminar-turbulent transition process. The low-Reynolds number terms can produce a delayed onset of the turbulent wall boundary layer and constitute therefore a very simple model for laminar-turbulent transition. In general, the use of the low-Reynolds number terms in the k-ω models is not recommended, and it is advised to use the more sophisticated, and more widely calibrated, models for laminar-turbulent transition instead.

4.4.1. Standard k-ω Model

4.4.1.1. Overview

The standard k-ω model in Ansys Fluent is based on a k-ω model proposed by Wilcox in [705] (p. 1094), which incorporates modifications for low-Reynolds number effects, compressibility, and shear flow spreading. One of the weak points of the 1998 Wilcox model is the sensitivity of the solutions to values for k and ω outside the shear layer (freestream sensitivity)., which can have a significant effect on the solution, especially for free shear flows [426] (p. 1077). There is a newer version of the model (Wilcox 2006 k-ω model [706] (p. 1094)), which did also not fully resolve the freestream sensitivity as shown in [426] (p. 1077) .

The standard k-ω model is an empirical model based on model transport equations for the turbulence kinetic energy (k) and the specific dissipation rate (ω), which can also be thought of as the ratio of ε to k [705] (p. 1094).

As the k-ω model has been modified over the years, production terms have been added to both the k and ω equations, which have improved the accuracy of the model for predicting free shear flows.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

4.4.1.2. Transport Equations for the Standard k-ω Model

The turbulence kinetic energy, k, and the specific dissipation rate, ω, are obtained from the following transport equations:

$$\frac{\partial}{\partial t}(\rho k)+\frac{\partial}{\partial x_i}(\rho k u_i)=\frac{\partial}{\partial x_j}\left(\Gamma_k\frac{\partial k}{\partial x_j}\right)+G_k-Y_k+S_k+G_b \tag{4.71}$$

and

$$\frac{\partial}{\partial t}(\rho \omega)+\frac{\partial}{\partial x_i}(\rho \omega u_i)=\frac{\partial}{\partial x_j}\left(\Gamma_\omega\frac{\partial \omega}{\partial x_j}\right)+G_\omega-Y_\omega+S_\omega+G_{\omega b} \tag{4.72}$$

In these equations, G_k represents the generation of turbulence kinetic energy due to mean velocity gradients. G_ω represents the generation of ω. Γ_k and Γ_ω represent the effective diffusivity of k and ω, respectively. Y_k and Y_ω represent the dissipation of k and ω due to turbulence. All of the above terms are calculated as described below. S_k and S_ω are user-defined source terms. G_b and $G_{\omega b}$ account for buoyancy terms as described in Effects of Buoyancy on Turbulence in the k-ω Models (p. 69).

4.4.1.3. Modeling the Effective Diffusivity

The effective diffusivities for the k-ω model are given by

$$\begin{aligned}\Gamma_k&=\mu+\frac{\mu_t}{\sigma_k}\\ \Gamma_\omega&=\mu+\frac{\mu_t}{\sigma_\omega}\end{aligned} \tag{4.73}$$

where σ_k and σ_ω are the turbulent Prandtl numbers for k and ω, respectively. The turbulent viscosity, μ_t, is computed by combining k and ω as follows:

$$\mu_t=\alpha^*\frac{\rho k}{\omega} \tag{4.74}$$

4.4.1.3.1. Low-Reynolds Number Correction

The coefficient α^* damps the turbulent viscosity causing a low-Reynolds number correction. It is given by

$$\alpha^*=\alpha^*_\infty\left(\frac{\alpha^*_0+Re_t/R_k}{1+Re_t/R_k}\right) \tag{4.75}$$

where

$$Re_t=\frac{\rho k}{\mu\omega} \tag{4.76}$$

$$R_k=6 \tag{4.77}$$

$$\alpha^*_0=\frac{\beta_i}{3} \tag{4.78}$$

$$\beta_i=0.072 \tag{4.79}$$

Note that in the high-Reynolds number form of the k-ω model, $\alpha^*=\alpha^*_\infty=1$.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

4.4.1.4. Modeling the Turbulence Production

4.4.1.4.1. Production of k

The term G_k represents the production of turbulence kinetic energy. From the exact equation for the transport of k, this term may be defined as

$$G_k = -\rho\overline{u_i' u_j'}\frac{\partial u_j}{\partial x_i} \tag{4.80}$$

To evaluate G_k in a manner consistent with the Boussinesq hypothesis,

$$G_k = \mu_t S^2 \tag{4.81}$$

where S is the modulus of the mean rate-of-strain tensor, defined in the same way as for the k-ε model (see Equation 4.62 (p. 57)).

4.4.1.4.2. Production of ω

The production of ω is given by

$$G_\omega = \alpha\frac{\omega}{k}G_k \tag{4.82}$$

where G_k is given by Equation 4.80 (p. 63).

The coefficient α is given by

$$\alpha = \frac{\alpha_\infty}{\alpha^*}\left(\frac{\alpha_0 + Re_t/R_\omega}{1 + Re_t/R_\omega}\right) \tag{4.83}$$

where R_ω = 2.95. α^* and Re_t are given by Equation 4.75 (p. 62) and Equation 4.76 (p. 62), respectively.

Note that in the high-Reynolds number form of the k-ω model, $\alpha = \alpha_\infty = 0.52$.

4.4.1.5. Modeling the Turbulence Dissipation

4.4.1.5.1. Dissipation of k

The dissipation of k is given by

$$Y_k = \rho\beta^* f_{\beta^*} k\omega \tag{4.84}$$

where

$$f_{\beta^*} = \begin{cases} 1 & \chi_k \le 0 \\ \frac{1+680\chi_k^2}{1+400\chi_k^2} & \chi_k > 0 \end{cases} \tag{4.85}$$

where

$$\chi_k \equiv \frac{1}{\omega^3}\frac{\partial k}{\partial x_j}\frac{\partial \omega}{\partial x_j} \tag{4.86}$$

and

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$\beta^* = \beta_i^* \left[1 + \zeta^* F(M_t) \right] \tag{4.87}$$

$$\beta_i^* = \beta_\infty^* \left(\frac{4/15 + (Re_t/R_\beta)^4}{1 + (Re_t/R_\beta)^4} \right) \tag{4.88}$$

$$\zeta^* = 1.5 \tag{4.89}$$

$$R_\beta = 8 \tag{4.90}$$

$$\beta_\infty^* = 0.09 \tag{4.91}$$

where Re_t is given by Equation 4.76 (p. 62).

4.4.1.5.2. Dissipation of ω

The dissipation of ω is given by

$$Y_\omega = \rho \beta f_\beta \omega^2 \tag{4.92}$$

where

$$f_\beta = \frac{1 + 70\chi_\omega}{1 + 80\chi_\omega} \tag{4.93}$$

$$\chi_\omega = \left| \frac{\Omega_{ij}\Omega_{jk}S_{ki}}{(\beta_\infty^* \omega)^3} \right| \tag{4.94}$$

$$\Omega_{ij} = \frac{1}{2} \left(\frac{\partial u_i}{\partial x_j} - \frac{\partial u_j}{\partial x_i} \right) \tag{4.95}$$

The strain rate tensor, S_{ij} is defined in Equation 4.25 (p. 47). Also,

$$\beta = \beta_i \left[1 - \frac{\beta_i^*}{\beta_i} \zeta^* F(M_t) \right] \tag{4.96}$$

β_i^* and $F(M_t)$ are defined by Equation 4.88 (p. 64) and Equation 4.97 (p. 64), respectively.

4.4.1.5.3. Compressibility Effects

The compressibility function, $F(M_t)$, is given by

$$F(M_t) = \begin{cases} 0 & M_t \le M_{t0} \\ M_t^2 - M_{t0}^2 & M_t > M_{t0} \end{cases} \tag{4.97}$$

where

$$M_t^2 \equiv \frac{2k}{a^2} \tag{4.98}$$

$$M_{t0} = 0.25 \tag{4.99}$$

$$a = \sqrt{\gamma RT} \tag{4.100}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Note that, in the high-Reynolds number form of the k-ω model, $\beta_i^*=\beta_\infty^*$. In the incompressible form, $\beta^*=\beta_i^*$.

Note:

The compressibility effects have been calibrated for a very limited number of free shear flow experiments, and it is not recommended for general use. It is disabled by default. For details, see Model Enhancements in the *Fluent User's Guide*.

4.4.1.6. Model Constants

$$\alpha_\infty^*=1,\ \alpha_\infty=0.52,\ \alpha_0=\frac{1}{9},\ \beta_\infty^*=0.09,\ \beta_i=0.072,\ R_\beta=8$$

$$R_k=6,\ R_\omega=2.95,\ \zeta^*=1.5,\ M_{t0}=0.25,\ \sigma_k=2.0,\ \sigma_\omega=2.0$$

4.4.2. Baseline (BSL) k-ω Model

4.4.2.1. Overview

The main problem with the Wilcox model is its well known strong sensitivity to freestream conditions. The baseline (BSL) k-ω model was developed by Menter [425] (p. 1077) to effectively blend the robust and accurate formulation of the k-ω model in the near-wall region with the freestream independence of the k-ε model in the far field. To achieve this, the k-ε model is converted into a k-ω formulation. The BSL k-ω model is similar to the standard k-ω model, but includes the following refinements:

- The standard k-ω model and the transformed k-ε model are both multiplied by a blending function and both models are added together. The blending function is designed to be one in the near-wall region, which activates the standard k-ω model, and zero away from the surface, which activates the transformed k-ε model.
- The BSL model incorporates a damped cross-diffusion derivative term in the ω equation.
- The modeling constants are different.

4.4.2.2. Transport Equations for the BSL k-ω Model

The BSL k-ω model has a similar form to the standard k-ω model:

$$\frac{\partial}{\partial t}(\rho k)+\frac{\partial}{\partial x_i}(\rho k u_i)=\frac{\partial}{\partial x_j}\left(\Gamma_k\frac{\partial k}{\partial x_j}\right)+G_k-Y_k+S_k+G_b \tag{4.101}$$

and

$$\frac{\partial}{\partial t}(\rho\omega)+\frac{\partial}{\partial x_i}(\rho\omega u_i)=\frac{\partial}{\partial x_j}\left(\Gamma_\omega\frac{\partial \omega}{\partial x_j}\right)+G_\omega-Y_\omega+D_\omega+S_\omega+G_{\omega b} \tag{4.102}$$

In these equations, the term G_k represents the production of turbulence kinetic energy, and is defined in the same manner as in the standard k-ω model. G_ω represents the generation of ω,

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

calculated as described in a section that follows. Γ_k and Γ_ω represent the effective diffusivity of k and ω, respectively, which are calculated as described in the section that follows. Y_k and Y_ω represent the dissipation of k and ω due to turbulence, calculated as described in Modeling the Turbulence Dissipation (p. 63). D_ω represents the cross-diffusion term, calculated as described in the section that follows. S_k and S_ω are user-defined source terms. G_b and $G_{\omega b}$ account for buoyancy terms as described in Effects of Buoyancy on Turbulence in the k-ω Models (p. 69).

4.4.2.3. Modeling the Effective Diffusivity

The effective diffusivities for the BSL k-ω model are given by

$$\Gamma_k = \mu + \frac{\mu_t}{\sigma_k} \tag{4.103}$$

$$\Gamma_\omega = \mu + \frac{\mu_t}{\sigma_\omega} \tag{4.104}$$

where σ_k and σ_ω are the turbulent Prandtl numbers for k and ω, respectively. The turbulent viscosity, μ_t, is computed as defined in Equation 4.74 (p. 62), and

$$\sigma_k = \frac{1}{F_1/\sigma_{k,1} + (1-F_1)/\sigma_{k,2}} \tag{4.105}$$

$$\sigma_\omega = \frac{1}{F_1/\sigma_{\omega,1} + (1-F_1)/\sigma_{\omega,2}} \tag{4.106}$$

The blending function F_1 is given by

$$F_1 = \tanh\left(\Phi_1^4\right) \tag{4.107}$$

$$\Phi_1 = \min\left[\max\left(\frac{\sqrt{k}}{0.09\omega y}, \frac{500\mu}{\rho y^2 \omega}\right), \frac{4\rho k}{\sigma_{\omega,2} D_\omega^+ y^2}\right] \tag{4.108}$$

$$D_\omega^+ = \max\left[2\rho\frac{1}{\sigma_{\omega,2}}\frac{1}{\omega}\frac{\partial k}{\partial x_j}\frac{\partial \omega}{\partial x_j}, 10^{-10}\right] \tag{4.109}$$

where y is the distance to the next surface and D_ω^+ is the positive portion of the cross-diffusion term (see Equation 4.117 (p. 67)).

4.4.2.4. Modeling the Turbulence Production

4.4.2.4.1. Production of k

The term G_k represents the production of turbulence kinetic energy, and is defined in the same manner as in the standard k-ω model. See Modeling the Turbulence Production (p. 63) for details.

4.4.2.4.2. Production of ω

The term G_ω represents the production of ω and is given by

$$G_\omega = \frac{\alpha\alpha^*}{\nu_t} G_k \tag{4.110}$$

Note that this formulation differs from the standard k-ω model (this difference is important for the SST model described in a later section). It also differs from the standard k-ω model in the way the term α_∞ is evaluated. In the standard k-ω model, α_∞ is defined as a constant (0.52). For the BSL k-ω model, α_∞ is given by

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$\alpha_{\infty}=F_1\alpha_{\infty,1}+\left(1-F_1\right)\alpha_{\infty,2} \tag{4.111}$$

where

$$\alpha_{\infty,1}=\frac{\beta_{i,1}}{\beta^*_{\infty}}-\frac{\kappa^2}{\sigma_{\omega,1}\sqrt{\beta^*_{\infty}}} \tag{4.112}$$

$$\alpha_{\infty,2}=\frac{\beta_{i,2}}{\beta^*_{\infty}}-\frac{\kappa^2}{\sigma_{\omega,2}\sqrt{\beta^*_{\infty}}} \tag{4.113}$$

where κ is 0.41.

4.4.2.5. Modeling the Turbulence Dissipation

4.4.2.5.1. Dissipation of k

The term Y_k represents the dissipation of turbulence kinetic energy, and is defined in a similar manner as in the standard k-ω model (see Modeling the Turbulence Dissipation (p. 63)). The difference is in the way the term f_{β^*} is evaluated. In the standard k-ω model, f_{β^*} is defined as a piecewise function. For the BSL k-ω model, f_{β^*} is a constant equal to 1. Thus,

$$Y_k=\rho\beta^* k\omega \tag{4.114}$$

4.4.2.5.2. Dissipation of ω

The term Y_ω represents the dissipation of ω, and is defined in a similar manner as in the standard k-ω model (see Modeling the Turbulence Dissipation (p. 63)). The difference is in the way the terms β_i and f_β are evaluated. In the standard k-ω model, β_i is defined as a constant (0.072) and f_β is defined in Equation 4.92 (p. 64). For the BSL k-ω model, f_β is a constant equal to 1. Thus,

$$Y_\omega=\rho\beta\omega^2 \tag{4.115}$$

Instead of having a constant value, β_i is given by

$$\beta_i=F_1\beta_{i,1}+\left(1-F_1\right)\beta_{i,2} \tag{4.116}$$

and F_1 is obtained from Equation 4.107 (p. 66).

Note that the constant value of 0.072 is still used for β_i in the low-Reynolds number correction for BSL to define α^*_0 in Equation 4.78 (p. 62).

4.4.2.6. Cross-Diffusion Modification

The BSL k-ω model is based on both the standard k-ω model and the standard k-ε model. To blend these two models together, the standard k-ε model has been transformed into equations based on k and ω, which leads to the introduction of a cross-diffusion term (D_ω in Equation 4.102 (p. 65)). D_ω is defined as

$$D_\omega=2\left(1-F_1\right)\rho\frac{1}{\omega\sigma_{\omega,2}}\frac{\partial k}{\partial x_j}\frac{\partial\omega}{\partial x_j} \tag{4.117}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

4.4.2.7. Model Constants

$$\sigma_{k,1}=2.0,\ \sigma_{\omega,1}=2.0,\ \sigma_{k,2}=1.0,\ \sigma_{\omega,2}=1.168$$

$$\beta_{i,1}=0.075,\beta_{i,2}=0.0828$$

All additional model constants (α^*_∞, α_∞, α_0, β^*_∞, R_β, R_k, R_ω, ζ^*, and M_{t0}) have the same values as for the standard k-ω model (see Model Constants (p. 65)).

4.4.3. Shear-Stress Transport (SST) k-ω Model

4.4.3.1. Overview

The SST k-ω model includes all the refinements of the BSL k-ω model, and in addition accounts for the transport of the turbulence shear stress in the definition of the turbulent viscosity.

These features make the SST k-ω model (Menter [425] (p. 1077)) more accurate and reliable for a wider class of flows (for example, adverse pressure gradient flows, airfoils, transonic shock waves) than the standard and the BSL k-ω models.

4.4.3.2. Modeling the Turbulent Viscosity

The BSL model described previously combines the advantages of the Wilcox and the k-ε model, but still fails to properly predict the onset and amount of flow separation from smooth surfaces. The main reason is that both models do not account for the transport of the turbulent shear stress. This results in an overprediction of the eddy-viscosity. The proper transport behavior can be obtained by a limiter to the formulation of the eddy-viscosity:

$$\mu_t=\frac{\rho k}{\omega}\frac{1}{\max\left[\frac{1}{\alpha^*},\frac{SF_2}{a_1\omega}\right]} \tag{4.118}$$

where S is the strain rate magnitude and α^* is defined in Equation 4.75 (p. 62). F_2 is given by

$$F_2=\tanh\left(\Phi_2^2\right) \tag{4.119}$$

$$\Phi_2=\max\left[2\frac{\sqrt{k}}{0.09\omega y},\frac{500\mu}{\rho y^2\omega}\right] \tag{4.120}$$

where y is the distance to the next surface.

4.4.3.3. Model Constants

$$\sigma_{k,1}=1.176,\ \sigma_{\omega,1}=2.0,\ \sigma_{k,2}=1.0,\ \sigma_{\omega,2}=1.168$$

$$a_1=0.31,\ \beta_{i,1}=0.075,\ \beta_{i,2}=0.0828$$

All additional model constants (α^*_∞, α_∞, α_0, β^*_∞, R_β, R_k, R_ω, ζ^*, and M_{t0}) have the same values as for the standard k-ω model (see Model Constants (p. 65)).

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

4.4.3.4. Treatment of the SST Model for Icing Simulations

An alternative SST roughness model has been implemented based on the Colebrook correlation by Aupoix [32] (p. 1054). As in the Spalart-Allmaras model, the concept of wall turbulent viscosity has been adopted, and it is estimated by modelling the wall values of k and ω.

Note that this modelling approach automatically disables the y^+-insensitive wall treatment which is outlined in y+-Insensitive Near-Wall Treatment for ω-based Turbulence Models (p. 152) at the respective wall.

Specifically, Aupoix proposed the following formulations to compute the non-dimensional wall values, k_w^+, ω_w^+ [32] (p. 1054):

$$k_w^+ = \max\left\{0, \frac{1}{\sqrt{\beta^*}} \tanh\left[\left(\frac{\ln\frac{h_s^+}{30}}{\ln 10} + 1 - \tanh\frac{h_s^+}{125}\right)\tanh\frac{h_s^+}{125}\right]\right\} \tag{4.121}$$

$$\omega_w^+ = \frac{300}{h_s^{+2}}\left(\tanh\frac{15}{4h_s^+}\right)^{-1} + \frac{191}{h_s^+}\left[1 - e^{\left(-\frac{h_s^+}{250}\right)}\right] \tag{4.122}$$

where $\beta^* = 0.09$, a standard constant in the SST k-ω model. y^+ and h_s^+ are defined as:

$$y^+ = \frac{\rho u_\tau d_w}{\mu} \tag{4.123}$$

$$h_s^+ = \frac{\rho u_\tau h_s}{\mu} \tag{4.124}$$

Therefore, all the wall values of k and ω are known:

$$k_w = f_w\left(u_\tau, k_w^+\right) = k_w^+ u_\tau^2 \tag{4.125}$$

$$\omega_w = f_w\left(u_\tau, \omega_w^+\right) = \frac{\rho \omega_w^+ u_\tau^2}{\mu} \tag{4.126}$$

4.4.4. Effects of Buoyancy on Turbulence in the k-ω Models

The effects of buoyancy can be included in the transport equations of the turbulence kinetic energy k (Equation 4.71 (p. 62), Equation 4.101 (p. 65)) and the specific dissipation rate ω (Equation 4.72 (p. 62), Equation 4.102 (p. 65)).

The turbulence generation due to buoyancy (G_b) is modeled in the same way as for the turbulence models based on the transport equation of the dissipation rate ε (see Effects of Buoyancy on Turbulence in the k-ε Models (p. 58)). It is by default included in the transport equation of the turbulence kinetic energy k.

The buoyancy term in the ω-equation ($G_{\omega b}$) is derived from the k and ε equations (Equation 4.39 (p. 50) and Equation 4.40 (p. 50)) using the following relations:

$$\varepsilon = \beta^* \omega k \tag{4.127}$$

$$\frac{D(\rho\varepsilon)}{Dt} = \beta^* \omega \frac{D(\rho k)}{DT} + \beta^* k \frac{D(\rho\omega)}{DT} \tag{4.128}$$

This derivation leads to the following transformation of the buoyancy source terms:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$G_{\omega b}=\frac{\omega}{k}C_{1\varepsilon}\left(C_{3\varepsilon}G_b\right)-\frac{\omega}{k}G_b \tag{4.129}$$

The first part of the buoyancy term from the ω equation, $G_{\omega b}$, comes from the transport equation of the dissipation rate. The model coefficient, $C_{1\varepsilon}$, is replaced with $(1+\alpha)$, where α is the corresponding coefficient of the production term in the ω-equation. In the BSL and SST model, this coefficient is a linear combination of the corresponding coefficients of the k-ω and the transformed k-ε models. For the k-ε model, the value of α is 0.44. The value of $C_{1\varepsilon}$ in the standard k-ε model is recovered from this value of α.

The coefficient $C_{3\varepsilon}$ is not specified, but is instead calculated according to Equation 4.67 (p. 59).

The final formulation of the buoyancy source terms for the ω-transport equation thus reads:

$$G_{\omega b}=\frac{\omega}{k}\left[(1+\alpha)C_{3\varepsilon}G_b-G_b\right] \tag{4.130}$$

The second part is included by default, whereas the first part is only included if the full buoyancy model is specified in the Viscous Models dialog box (see Including Buoyancy Effects on Turbulence in the *Fluent User's Guide*).

4.4.5. Turbulence Damping

For details about turbulence damping, see Turbulence Damping (p. 167).

4.5. Generalized k-ω (GEKO) Model

This section describes the theory behind the GEKO model. Information is presented in the following sections:

4.5.1. Model Formulation

4.5.2. Limitations

For details about using the model in Ansys Fluent, see Modeling Turbulence and Setting up the Generalized k-ω (GEKO) Model in the User's Guide.

Within the RANS concept, it is not possible to cover all flows with sufficient accuracy using a single model. Therefore, it is necessary that industrial CFD codes offer a multitude of turbulence models, allowing users to select the best model for their application. However, this is not an optimal strategy, as not all models are of the same quality (in terms of robustness, interoperability with other models, near wall treatment). Switching from one model to another has therefore additional consequences beyond the desired change in the solution. An alternative approach is to offer a single model with enough flexibility to cover a wide range of applications.

The goal of the GEKO model is to consolidate the two-equation models into such a formulation. The model provides free parameters that you can adjust for specific types of applications, without negative impact on the basic calibration of the model. This is different from classical models, where the coefficients of the model are provided (for example, $C_{1\varepsilon}$ and $C_{2\varepsilon}$ in $k-\varepsilon$ models [338] (p. 1072)) but can hardly be changed because they are inherently intertwined and any change would typically lead to a loss of calibration with the most basic flows like a flat plate boundary layer.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

4.5.1. Model Formulation

The GEKO model is currently not published. It is based on a k-ω formulation and features four free coefficients that can be tuned/optimized within given limits without negative effect on the underlying calibration for wall boundary layers at zero pressure gradient as well as channel and pipe flows. The coefficients are:

- C_{SEP} - Parameter to optimize flow separation from smooth surfaces.
  - 0.7<C_{SEP}<2.5 (default C_{SEP} = 1.75)
  - Increasing C_{SEP} leads to earlier and stronger separation.
  - C_{SEP}=1 mimics the standard $k-\varepsilon$ model, C_{SEP}=1.75 is close in performance to the SST model.
- C_{NW} - Parameter to optimize flow in non-equilibrium near wall regions (such as heat transfer or C_f).
  - -2<C_{NW}<2 (default C_{NW}=0.5)
  - Increasing C_{NW} leads to higher heat transfer rates at reattachment locations (backstep case, for example).
  - C_{NW} is mostly used to adjust for flows with heat transfer in impingement zones. It has a very strong default value and typically should not be changed except if detailed experimental data are available.
- C_{MIX} - Parameter to optimize strength of mixing in free shear flows.
  - 0<C_{MIX}<1 (default - correlation $C_{MixCor}=0.35\mathrm{sgn}(C_{SEP}-1)\sqrt{|C_{SEP}-1|}$)
  - Increasing C_{MIX} from its default value will lead to stronger mixing in free shear flows (mixing layer). The correlation C_{MixCor} assures that, for changes in C_{SEP}, the classical mixing layer spreading rate is preserved. However, C_{MIX} can also be given a value other than the correlation.
- C_{JET} - Parameter to optimize free shear layer mixing (optimize free jets independent of mixing layer).
  - 0<C_{JET}<1 (default C_{JET}=0.9).
  - Increasing C_{MIX} will also increase the spreading rates of free jet flows. This can be undesirable, and therefore C_{JET} allows you to reduce the effect of C_{MIX} on free jet flows. C_{JET} is only active when C_{MIX} is non-zero and reduces the spreading rates of free jets.
  - This parameter will only be required in a limited number of applications. C_{JET} is a sub-model of C_{MIX} – this means that when C_{MIX}=0, C_{JET} will also not be active.

You also have the option to tune the model with the curvature correction parameter, C_{curv}. This parameter is also available for other models and is described in Curvature Correction for the Spalart-Allmaras and Two-Equation Models (p. 162).

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

There are parameter combinations that are of specific interest. For C_{SEP}=1, C_{MIX}=0, and C_{NW}=1, the model obtains an exact transformation to the standard $k-\varepsilon$ model (albeit with the improved near wall treatment of the $k-\omega$ model). Note that C_{JET} is not relevant as C_{MIX}=0.

The default values of C_{SEP}=1.75, C_{MIX}=0.3, C_{NW}=0.5, and C_{JET}=0.9 give a close approximation of the SST model (especially in terms of separation behavior not in terms of formulation). For free shear flows, this combination is superior compared to SST. It gives similar spreading rates for mixing layers but improved spreading for round jets.

By default, the GEKO model features a realizability limiter that ensures positive normal stresses in the entire domain:

$$\nu_t=\min\left(\frac{k}{\omega},C_{REAL}\frac{k}{S}\right)=\frac{k}{\max(\omega,S/C_{REAL})};\ C_{REAL}=0.577 \tag{4.131}$$

The function involving C_{MIX} and C_{JET} is controlled by a blending function F_{Blend} which deactivates these parameters inside boundary layers (F_{Blend}=1). For free flows, F_{Blend}=0.

$$F_{MIX}=F(C_{MIX},C_{JET})(1-F_{Blend}) \tag{4.132}$$

This function requires wall-distance, which can be expensive to compute in some cases (flows with moving geometries, changing meshes). For such applications, the model can be run with **Wall Distance Free** enabled. In this mode, the parameters C_{MIX} and C_{JET} are de-activated and F_{Blend} is automatically set to 1. To maintain free mixing layer spreading rates, when using the wall distance free option, you should select the $k-\varepsilon$ model variant (C_{SEP}=1, C_{MIX}=0, C_{NW}=1). However, more aggressive settings for separation prediction are possible (C_{SEP}>1) but would result in lower spreading rates for free shear flows. It is important to understand that the wall distance can also be required by other models (for example, transition models, hybrid RANS-LES models). If you select **Wall Distance Free**, then all other model options using wall distance become unavailable. If you first select a sub-model within the turbulence dialog box that requires wall-distance, then the **Wall Distance Free** option becomes unavailable.

C_{SEP}, C_{NW}, C_{MIX}, and F_{Blend} can also be specified as an expression or via UDF (see DEFINE_KW_GEKO Coefficients and Blending Function in the *Fluent Customization Manual*).

Beside the four free coefficients and F_{Blend}, four auxiliary parameters are available which allow further fine-tuning of the model behavior by changing their constant values. The majority of users need not modify the default values of these coefficients:

C_{NW_SUB}: allows the adjustment of log-layer and c_f (default = 1.7).

C_{JET_AUX}: allows fine-tuning of the parameter to optimize free jets (default = 2.0). Higher values make the impact of C_{JET} more effective.

C_{BF_TUR}: Factor for main shielding function F_{Blend} (default = 2.0). Decreasing this value will decrease the thickness of the layer near walls, where C_{MIX} and C_{JET} are deactivated. This will result in the activation of the C_{MIX} and C_{JET} formulation closer to walls.

C_{BF_LAM}: Only active if you have enabled the Intermittency Transition Model. Factor is a part of F_{Blend} and shields the laminar boundary layer (default = 25.0). This parameter provides additional shielding

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

of the boundary layer from any impact of C_{MIX} and C_{JET} in laminar regions. This parameter is required to avoid any influence of these parameters when the GEKO model is combined with a laminar-turbulent transition model. Setting a lower value will reduce this shielding.

4.5.2. Limitations

The GEKO implementation has the following limitations:

- Combination of GEKO with multiphase is not yet validated. However, the GEKO model is not fundamentally different from existing two-equation models to expect model-specific difficulties.

Caution:

All possible combinations outside the turbulence panel have not been covered. In the situation where a sub-model (for example, in multi-phase flows) requires wall distance, it is recommended to avoid the **Wall Distance Free** option on the turbulence dialog box.

4.6. k-kl-ω Transition Model

This section describes the theory behind the k-kl-ω Transition model. For details about using the model in Ansys Fluent, see Modeling Turbulence and Setting Up the Transition k-kl-ω Model in the User's Guide.

For more information, see the following sections:

4.6.1. Overview

4.6.2. Transport Equations for the k-kl-ω Model

4.6.1. Overview

The k-kl-ω transition model [682] (p. 1092) is used to predict boundary layer development and calculate transition onset. This model can be used to effectively address the transition of the boundary layer from a laminar to a turbulent regime.

4.6.2. Transport Equations for the k-kl-ω Model

The k-kl-ω model is considered to be a three-equation eddy-viscosity type, which includes transport equations for turbulent kinetic energy (k_T), laminar kinetic energy (k_L), and the inverse turbulent time scale (ω)

$$\frac{Dk_T}{Dt} = P_{K_T} + R + R_{NAT} - \omega k_T - D_T + \frac{\partial}{\partial x_j}\left[\left(\nu + \frac{\alpha_T}{\alpha_k}\right)\frac{\partial k_T}{\partial x_j}\right] \tag{4.133}$$

$$\frac{Dk_L}{Dt} = P_{K_L} - R - R_{NAT} - D_L + \frac{\partial}{\partial x_j}\left[\nu\frac{\partial k_L}{\partial x_j}\right] \tag{4.134}$$

$$\begin{aligned}\frac{D_\omega}{D_t} = \;& C_{\omega 1}\frac{\omega}{k_T}P_{k_T} + \left(\frac{C_{\omega R}}{f_W} - 1\right)\frac{\omega}{k_T}\left(R + R_{NAT}\right) - C_{\omega 2}\omega^2 \\ & + C_{\omega 3} f_\omega \alpha_T f_W^2 \frac{\sqrt{k_T}}{d^3} + \frac{\partial}{\partial x_j}\left[\left(\nu + \frac{\alpha_T}{\alpha_\omega}\right)\frac{\partial \omega}{\partial x_j}\right]\end{aligned} \tag{4.135}$$

The inclusion of the turbulent and laminar fluctuations on the mean flow and energy equations via the eddy viscosity and total thermal diffusivity is as follows:

$$\overline{-u_i u_j} = \nu_{TOT}\left(\frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i}\right) - \frac{2}{3}k_{TOT}\delta_{ij} \tag{4.136}$$

$$\overline{-u_i \theta} = \alpha_{\theta,TOT}\frac{\partial \theta}{\partial x_i} \tag{4.137}$$

The effective length is defined as

$$\lambda_{eff} = MIN\left(C_\lambda d, \lambda_T\right) \tag{4.138}$$

where λ_T is the turbulent length scale and is defined by

$$\lambda_T = \frac{\sqrt{k}}{\omega} \tag{4.139}$$

and the small scale energy is defined by

$$k_{T,s} = f_{ss} f_W k_T \tag{4.140}$$

$$f_W = \frac{\lambda_{eff}}{\lambda_T} \tag{4.141}$$

$$f_{ss} = exp\left[-\left(\frac{C_{ss}\nu\Omega}{k_T}\right)^2\right] \tag{4.142}$$

The large scale energy is given by

$$k_{T,l} = k_T - k_{T,s} \tag{4.143}$$

Note that the sum of Equation 4.140 (p. 74) and Equation 4.143 (p. 74) yields the turbulent kinetic energy k_T.

The turbulence production term generated by turbulent fluctuations is given by

$$P_{k_T} = \nu_{T,s} S^2 \tag{4.144}$$

where the small-scale turbulent viscosity is $\nu_{T,s}$

$$\nu_{T,s} = f_\nu f_{INT} C_\mu \sqrt{k_{T,s}}\lambda_{eff} \tag{4.145}$$

and

$$C_\mu = \frac{1}{A_0 + A_s\left(S/\omega\right)} \tag{4.146}$$

$$f_\nu = 1 - exp\left(-\frac{\sqrt{Re_{T,s}}}{A_\nu}\right) \tag{4.147}$$

A damping function defining the turbulent production due to intermittency is given by

$$f_{INT} = MIN\left(\frac{k_L}{C_{INT}k_{TOT}}, 1\right) \tag{4.148}$$

$$Re_{T,s} = \frac{f_W^2 k_T}{\nu\omega} \tag{4.149}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

In Equation 4.134 (p. 73), P_{k_L} is the production of laminar kinetic energy by large scale turbulent fluctuations, such that

$$P_{k_L}=\nu_{T,l}S^2 \tag{4.150}$$

The large-scale turbulent viscosity $\nu_{T,1}$ is modeled as

$$\nu_{T,1}=MIN\left\{\nu^*_{T,1},\frac{0.5\left(k_L+k_{T,1}\right)}{S}\right\} \tag{4.151}$$

where

$$\nu^*_{T,1}=f_{\tau,1}C_{11}\left(\frac{\Omega\lambda^2_{eff}}{\nu}\right)\sqrt{k_{T,1}}\lambda_{eff}+\beta_{TS}C_{12}\varphi_{NAT}d^2\Omega \tag{4.152}$$

The limit in Equation 4.151 (p. 75) binds the realizability such that it is not violated in the two-dimensional developing boundary layer. The time-scale-based damping function $f_{\tau,1}$ is

$$f_{\tau,1}=1-exp\left[-C_{\tau,1}\frac{k_{T,1}}{\lambda^2_{eff}\Omega^2}\right] \tag{4.153}$$

where β_{TS} from Equation 4.152 (p. 75) is

$$\beta_{TS}=1-exp\left(-\frac{MAX\left(\varphi_{NAT}-C_{TS,crit},0\right)^2}{A_{TS}}\right) \tag{4.154}$$

$$\varphi_{NAT}=\frac{d^2\Omega}{\nu} \tag{4.155}$$

Near-wall dissipation is given by

$$D_T=2\nu\frac{\partial\sqrt{k_T}}{\partial x_j}\frac{\partial\sqrt{k_T}}{\partial x_j} \tag{4.156}$$

$$D_L=2\nu\frac{\partial\sqrt{k_L}}{\partial x_j}\frac{\partial\sqrt{k_L}}{\partial x_j} \tag{4.157}$$

In Equation 4.133 (p. 73) – Equation 4.135 (p. 73), R represents the averaged effect of the breakdown of streamwise fluctuations into turbulence during bypass transition:

$$R=C_R\beta_{BP}k_L\omega/\ f_W \tag{4.158}$$

β_{BP}, which is the threshold function controls the bypass transition process:

$$\beta_{BP}=1-exp\left(-\frac{\varphi_{BP}}{A_{BP}}\right) \tag{4.159}$$

$$\varphi_{BP}=MAX\left[\left(\frac{k_T}{\nu\Omega}-C_{BP,crit}\right),0\right] \tag{4.160}$$

The breakdown to turbulence due to instabilities is considered to be a natural transition production term, given by

$$R_{NAT}=C_{R,NAT}\beta_{NAT}k_L\Omega \tag{4.161}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$\beta_{NAT}=1-exp\left[-\frac{MAX\left(\varphi_{NAT}-C_{NAT,crit}/\ f_{NAT,crit},0\right)}{A_{NAT}}\right] \tag{4.162}$$

$$f_{NAT,crit}=1-exp\left(C_{NC}\frac{\sqrt{k_L}d}{\nu}\right) \tag{4.163}$$

The use of ω as the scale-determining variable can lead to a reduced intermittency effect in the outer region of a turbulent boundary layer, and consequently an elimination of the wake region in the velocity profile. From Equation 4.135 (p. 73), the following damping is defined as

$$f_{\omega}=1-exp\left[-0.41\left(\frac{\lambda_{eff}}{\lambda_T}\right)^4\right] \tag{4.164}$$

The total eddy viscosity and eddy diffusivity included in Equation 4.136 (p. 74) and Equation 4.137 (p. 74) are given by

$$\nu_{TOT}=\nu_{T,s}+\nu_{T,l} \tag{4.165}$$

$$\alpha_{\theta,TOT}=f_W\left(\frac{k_T}{k_{TOT}}\right)\frac{\nu_{T,s}}{Pr_{\theta}}+\left(1-f_W\right)C_{\alpha,\theta}\sqrt{k_T}\lambda_{eff} \tag{4.166}$$

The turbulent scalar diffusivity in Equation 4.133 (p. 73) – Equation 4.135 (p. 73) is defined as

$$\alpha_T=f_{\nu}C_{\mu,std}\sqrt{k_{T,s}}\lambda_{eff} \tag{4.167}$$

$$k_{TOT}=k_T+k_L \tag{4.168}$$

A compressibility effects option, similar to the one in the k-ε model (Effects of Compressibility on Turbulence in the k-ε Models (p. 59)) is available for the k-kl-ω model. By default, this compressibility effects option is turned off. For details see, Model Enhancements in the *Fluent User's Guide*.

4.6.2.1. Model Constants

The model constants for the k-kl-ω transition model are listed below [682] (p. 1092)

$$A_0=4.04,\ A_s=2.12,\ A_v=6.75,\ A_{BP}=0.6$$

$$A_{NAT}=200,\ A_{TS}=200,\ C_{BP,crit}=1.2,\ C_{NC}=0.1$$

$$C_{NAT,crit}=1250,\ C_{INT}=0.75,\ C_{TS,crit}=1000,\ C_{R,NAT}=0.02$$

$$C_{11}=3.4\times10^{-6},\ C_{12}=1.0\times10^{-10},\ C_R=0.12,\ C_{\alpha,\theta}=0.035$$

$$C_{SS}=1.5,\ C_{\tau,1}=4360,C_{\omega1}=0.44,\ C_{\omega2}=0.92$$

$$C_{\omega3}=0.3,\ C_{\omega R}=1.5,\ C_{\lambda}=2.495,\ C_{\mu,std}=0.09$$

$$Pr_{\theta}=0.85,\ \sigma_k=1,\ \sigma_{\omega}=1.17$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

4.7. Transition SST Model

This section describes the theory behind the Transition SST model. Information is presented in the following sections:

4.7.1. Overview

4.7.2. Transport Equations for the Transition SST Model

4.7.3. Mesh Requirements

4.7.4. Specifying Inlet Turbulence Levels

For details about using the model in Ansys Fluent, see Modeling Turbulence and Setting Up the Transition SST Model in the User's Guide.

4.7.1. Overview

The Transition SST model (also known as the γ-Re_θ model) is based on the coupling of the SST $k-\omega$ transport equations with two other transport equations, one for the intermittency and one for the transition onset criteria, in terms of momentum-thickness Reynolds number. An ANSYS empirical correlation (Langtry and Menter) has been developed to cover standard bypass transition as well as flows in low freestream turbulence environments.

In addition, a very powerful option has been included to allow you to enter your own user-defined empirical correlation, which can then be used to control the transition onset momentum thickness Reynolds number equation.

Note the following limitations:

- The Transition SST model is only applicable to wall-bounded flows. Like all other engineering transition models, the model is not applicable to transition in free shear flows. The model will predict free shear flows as fully turbulent.
- The Transition SST model is not Galilean invariant and should therefore not be applied to surfaces that move relative to the coordinate system for which the velocity field is computed; for such cases the Intermittency Transition model should be used instead.
- The Transition SST model is designed for flows with a defined nonzero freestream velocity (that is, the classical boundary layer situation). It is not suitable for fully developed pipe / channel flows where no freestream is present. For the same reason, it is also not suitable for wall jet flows. For such scenarios, the Intermittency Transition model should be used instead. Note, however, that it might be necessary to adjust the Intermittency Transition model for such flows by modifying the underlying correlations.
- The Transition SST model has not been calibrated in combination with other physical effects that affect the source terms of the turbulence model, such as:
  - buoyancy
  - multiphase turbulence

To learn how to set up the Transition SST model, see Setting Up the Transition SST Model (in the User's Guide).

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

4.7.2. Transport Equations for the Transition SST Model

The transport equation for the intermittency γ is defined as:

$$\frac{\partial(\rho\gamma)}{\partial t}+\frac{\partial(\rho U_j\gamma)}{\partial x_j}=P_{\gamma1}-E_{\gamma1}+P_{\gamma2}-E_{\gamma2}+\frac{\partial}{\partial x_j}\left[\left(\mu+\frac{\mu_t}{\sigma_\gamma}\right)\frac{\partial\gamma}{\partial x_j}\right] \tag{4.169}$$

The transition sources are defined as follows:

$$\begin{aligned} P_{\gamma1} &= C_{a1}F_{length}\rho S\left[\gamma F_{onset}\right]^{c_{\gamma3}} \\ E_{\gamma1} &= C_{e1}P_{\gamma1}\gamma \end{aligned} \tag{4.170}$$

where S is the strain rate magnitude, F_{length} is an empirical correlation that controls the length of the transition region, and C_{a1} and C_{e1} hold the values of 2 and 1, respectively. The destruction/re-laminarization sources are defined as follows:

$$\begin{aligned} P_{\gamma2} &= C_{a2}\rho\Omega\gamma F_{turb} \\ E_{\gamma2} &= C_{e2}P_{\gamma2}\gamma \end{aligned} \tag{4.171}$$

where Ω is the vorticity magnitude. The transition onset is controlled by the following functions:

$$\begin{aligned} Re_V &= \frac{\rho y^2 S}{\mu} \\ R_T &= \frac{\rho k}{\mu\omega} \end{aligned} \tag{4.172}$$

$$F_{onset1} = \frac{Re_v}{2.193Re_{\theta c}} \tag{4.173}$$

$$\begin{aligned} F_{onset2} &= min\left(max\left(F_{onset1}, F_{onset1}^4\right), 2.0\right) \\ F_{onset3} &= max\left(1-\left(\frac{R_T}{2.5}\right)^3, 0\right) \\ F_{onset} &= max\left(F_{onset2}-F_{onset3}, 0\right) \\ F_{turb} &= e^{-\left(\frac{R_T}{4}\right)^4} \end{aligned} \tag{4.174}$$

where y is the wall distance and $Re_{\theta c}$ is the critical Reynolds number where the intermittency first starts to increase in the boundary layer. This occurs upstream of the transition Reynolds number $\tilde{Re}_{\theta t}$ and the difference between the two must be obtained from an empirical correlation. Both the F_{length} and $Re_{\theta c}$ correlations are functions of $\tilde{Re}_{\theta t}$.

The constants for the intermittency equation are:

$$C_{a1}=2; C_{e1}=1; C_{a2}=0.06; C_{e2}=50; c_{\gamma3}=0.5; \sigma_\gamma=1.0 \tag{4.175}$$

The transport equation for the transition momentum thickness Reynolds number $\tilde{Re}_{\theta t}$ is

$$\frac{\partial\left(\rho\tilde{Re}_{\theta t}\right)}{\partial t}+\frac{\partial\left(\rho U_j\tilde{Re}_{\theta t}\right)}{\partial x_j}=P_{\theta t}+\frac{\partial}{\partial x_j}\left[\sigma_{\theta t}\left(\mu+\mu_t\right)\frac{\partial\tilde{Re}_{\theta t}}{\partial x_j}\right] \tag{4.176}$$

The source term is defined as follows:

$$\begin{aligned} P_{\theta t} &= c_{\theta t}\frac{\rho}{t}\left(Re_{\theta t}-\tilde{Re}_{\theta t}\right)\left(1.0-F_{\theta t}\right) \\ t &= \frac{500\mu}{\rho U^2} \end{aligned} \tag{4.177}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$F_{\theta t}=min\left(max\left(F_{wake}e^{\left(-\frac{y}{\delta}\right)^4},1.0-\left(\frac{\gamma-1/\ 50}{1.0-1/\ 50}\right)^2\right),1.0\right) \tag{4.178}$$

$$\begin{aligned} \theta_{BL} &= \frac{\tilde{Re_{\theta t}}\mu}{\rho U} \\ \delta_{BL} &= \frac{15}{2}\theta_{BL} \\ \delta &= \frac{50\Omega y}{U}\delta_{BL} \end{aligned} \tag{4.179}$$

$$\begin{aligned} Re_{\omega} &= \frac{\rho\omega y^2}{\mu} \\ F_{wake} &= e^{-\left(\frac{Re_{\omega}}{1E+5}\right)^2} \end{aligned} \tag{4.180}$$

The model constants for the $\tilde{Re_{\theta t}}$ equation are:

$$c_{\theta t}=0.03;\quad \sigma_{\theta t}=2.0 \tag{4.181}$$

The boundary condition for $\tilde{Re_{\theta t}}$ at a wall is zero flux. The boundary condition for $\tilde{Re_{\theta t}}$ at an inlet should be calculated from the empirical correlation based on the inlet turbulence intensity.

The model contains three empirical correlations. $Re_{\theta t}$ is the transition onset as observed in experiments. This has been modified from Menter et al. [430] (p. 1078) in order to improve the predictions for natural transition. It is used in Equation 4.176 (p. 78). F_{length} is the length of the transition zone and is substituted in Equation 4.169 (p. 78). $Re_{\theta c}$ is the point where the model is activated in order to match both $Re_{\theta t}$ and F_{length}, and is used in Equation 4.173 (p. 78). These empirical correlations are provided by Langtry and Menter [333] (p. 1072).

$$\begin{aligned} Re_{\theta t} &= f\left(Tu,\lambda\right) \\ F_{length} &= f\left(\tilde{Re_{\theta t}}\right) \\ Re_{\theta c} &= f\left(\tilde{Re_{\theta t}}\right) \end{aligned} \tag{4.182}$$

The first empirical correlation is a function of the local turbulence intensity, Tu:

$$Tu=\frac{100}{U}\sqrt{\frac{2}{3}k} \tag{4.183}$$

where k is the turbulent energy.

The Thwaites' pressure gradient coefficient λ_θ is defined as

$$\lambda_\theta=\left(\theta^2/\ \nu\right)dU/\ ds \tag{4.184}$$

where $dU/\ ds$ is the acceleration in the streamwise direction.

4.7.2.1. Separation-Induced Transition Correction

The modification for separation-induced transition is:

$$\begin{aligned} \gamma_{sep} &= min\left(C_{s1}max\left[\left(\frac{Re_v}{3.235Re_{\theta c}}\right)-1,0\right]F_{reattch},2\right)\ F_{\theta t} \\ F_{reattch} &= e^{-\left(\frac{R_T}{20}\right)^4} \\ \gamma_{eff} &= max\left(\gamma,\gamma_{sep}\right) \end{aligned} \tag{4.185}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Here, C_{s1} is a constant with a value of 2.

The model constants in Equation 4.185 (p. 79) have been adjusted from those of Menter et al. [430] (p. 1078) in order to improve the predictions of separated flow transition. The main difference is that the constant that controls the relation between Re_v and $Re_{\theta c}$ was changed from 2.193, its value for a Blasius boundary layer, to 3.235, the value at a separation point where the shape factor is 3.5 [430] (p. 1078). The boundary condition for γ at a wall is zero normal flux, while for an inlet, γ is equal to 1.0.

4.7.2.2. Coupling the Transition Model and SST Transport Equations

The transition model interacts with the SST turbulence model by modification of the k-equation (Equation 4.101 (p. 65)), as follows:

$$\frac{\partial}{\partial t}(\rho k)+\frac{\partial}{\partial x_i}(\rho k u_i)=\frac{\partial}{\partial x_j}\left(\Gamma_k\frac{\partial k}{\partial x_j}\right)+G_k^*-Y_k^*+S_k \tag{4.186}$$

$$G_k^*=\gamma_{eff}\tilde{G}_k \tag{4.187}$$

$$Y_k^*=min\left(max\left(\gamma_{eff},0.1\right),1.0\right)Y_k \tag{4.188}$$

where $\tilde{G}_k$ and Y_k are the original production and destruction terms for the SST model. Note that the production term in the ω-equation is not modified. The rationale behind the above model formulation is given in detail in Menter et al. [430] (p. 1078).

In order to capture the laminar and transitional boundary layers correctly, the mesh must have a y^+ of approximately one. If the y^+ is too large (that is, > 5), then the transition onset location moves upstream with increasing y^+. It is recommended that you use the bounded second order upwind based discretization for the mean flow, turbulence and transition equations.

4.7.2.3. Transition SST and Rough Walls

When the Transition SST Model is used together with rough walls, the roughness correlation must be enabled in the Viscous Model Dialog Box. This correlation requires the geometric roughness height K as an input parameter, since, for the transition process from laminar to turbulent flow, the geometric roughness height is more important than the equivalent sand-grain roughness height K_s. Guidance to determine the appropriate equivalent sand-grain roughness height (based on the geometric roughness height, shape and distribution of the roughness elements) can be obtained, for example, from Schlichting and Gersten [575] (p. 1086) and Coleman et al. [117] (p. 1059).

The roughness correlation is a modification of the built-in correlation for $\widetilde{\mathrm{Re}}_{\theta t}$ and is defined as:

$$\mathrm{Re}_{\theta t,rough}=\widetilde{\mathrm{Re}}_{\theta t}\cdot f(K) \tag{4.189}$$

The new defined $\mathrm{Re}_{\theta t,rough}$ is then used in the correlations for F_{length} and $\mathrm{Re}_{\theta c}$. $\widetilde{\mathrm{Re}}_{\theta t}$ represents the transition momentum thickness Reynolds number. The value specified for the geometric roughness will apply to all walls. In case a different value is required for different walls, a user-defined function can be specified. It is important to note that the function for K is used in the volume (not at the wall). The function therefore must cover the region of the boundary layer and beyond where it should be applied. As an example, assume a roughness strip on a flat plate (x-streamwise direction, y wall normal z-spanwise) at the location $1<x<1.2$ (m) and a boundary layer thickness in that region

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

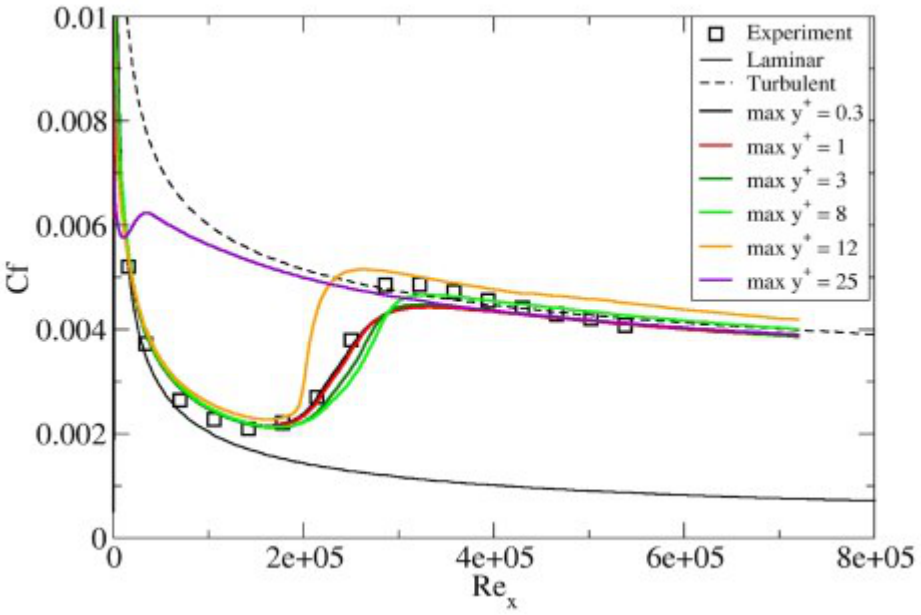
of perhaps $\delta=0.01\mathrm{m}$ (spanwise extent $-0.1<z<0.1$ (m)). The following pseudo-code will switch between roughness K0 everywhere (could be zero) and K1 at the transition strip. The height in the y-direction does not have to be exactly the boundary layer thickness – it can be much larger – as long as it does not impact other walls in the vicinity.

$$\begin{aligned} &fstep=0 \\ &if\left(\left(1<x<1.2\right)\ and\ \left(0<y<0.02\right)\ and\ \left(-0.1<z<0.1\right)\right)fstep=1 \\ &K=K0\cdot\left(1-fstep\right)+K1\cdot fstep \end{aligned} \tag{4.190}$$

4.7.3. Mesh Requirements

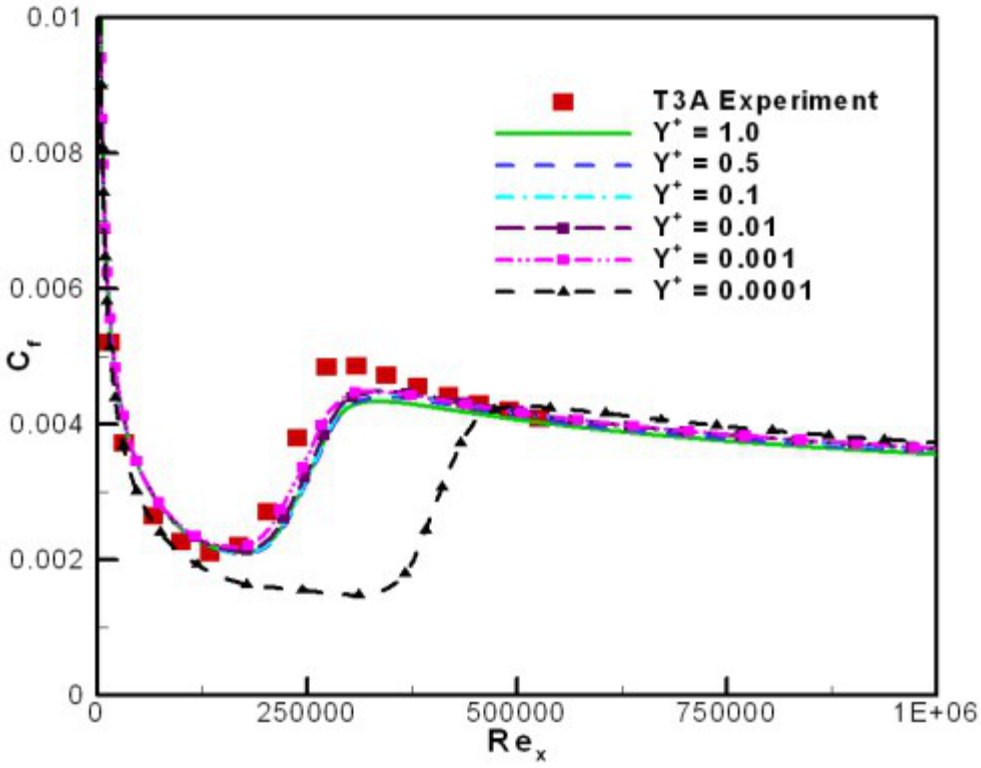
The effect of increasing and decreasing y^+ for a flat plate test case (T3A) is shown in Figure 4.1: Effect of Increasing y+ for the Flat Plate T3A Test Case (p. 81) and Figure 4.2: Effect of Decreasing y+ for the Flat Plate T3A Test Case (p. 82). For y^+ values between 0.001 and 1, there is almost no effect on the solution. Once the maximum y^+ increases above 8, the transition onset location begins to move upstream. At a maximum y^+ of 25, the boundary layer is almost completely turbulent. For y^+ values below 0.001, the transition location appears to move downstream. This is presumably caused by the large surface value of the specific turbulence frequency ω, which scales with the first mesh node height. For these reasons, very small y^+ values (below 0.001) should be avoided.

Figure 4.1: Effect of Increasing y+ for the Flat Plate T3A Test Case



Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

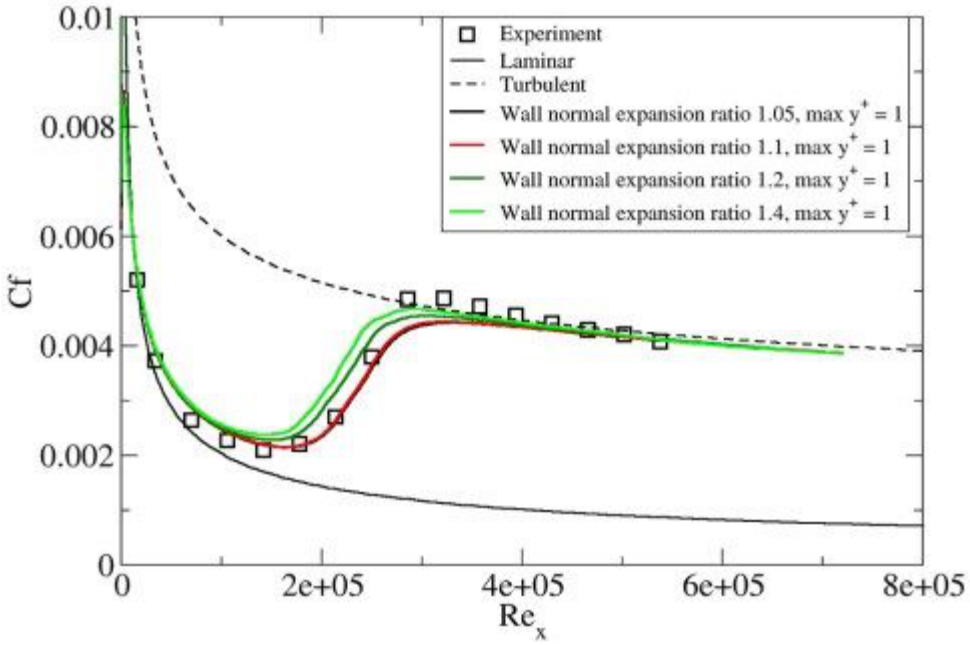
Figure 4.2: Effect of Decreasing y+ for the Flat Plate T3A Test Case



The effect of the wall normal expansion ratio from a y^+ value of 1 is shown in Figure 4.3: Effect of Wall Normal Expansion Ratio for the Flat Plate T3A Test Case (p. 83). For expansion factors of 1.05 and 1.1, there is no effect on the solution. For larger expansion factors of 1.2 and 1.4, there is a small but noticeable upstream shift in the transition location. Because the sensitivity of the solution to wall-normal mesh resolution can increase for flows with pressure gradients, it is recommended that you apply meshes with $y^+<1$ and expansion factors smaller than 1.1.

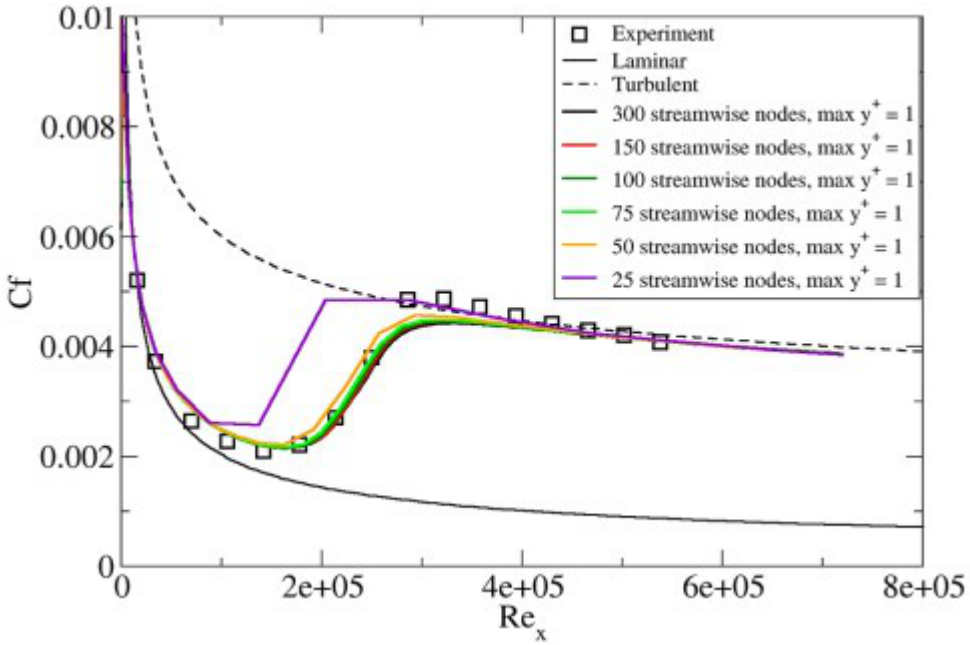
Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Figure 4.3: Effect of Wall Normal Expansion Ratio for the Flat Plate T3A Test Case



The effect of streamwise mesh refinement is shown in Figure 4.4: Effect of Streamwise Mesh Density for the Flat Plate T3A Test Case (p. 83). Surprisingly, the model was not very sensitive to the number of streamwise nodes. The solution differed significantly from the mesh-independent one only for the case of 25 streamwise nodes where there was only one cell in the transitional region. Nevertheless, the mesh-independent solution appears to occur when there is approximately 75–100 streamwise mesh nodes on the flat plate. Also, separation-induced transition occurs over a very short length; for cases where this is important, a fine mesh is necessary.

Figure 4.4: Effect of Streamwise Mesh Density for the Flat Plate T3A Test Case



Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

One point to note is that for sharp leading edges, often transition can occur due to a small leading edge separation bubble. If the mesh is too coarse, the rapid transition caused by the separation bubble is not captured.

Based on the mesh sensitivity study, the recommended best practice mesh guidelines are a max y^+ of 1, a wall normal expansion ratio that is less than 1.1, and about 75–100 mesh nodes in the streamwise direction. Note that if separation-induced transition is present, additional mesh nodes in the streamwise direction are most likely needed. For a turbine blade, that would translate into 100–150 cells in the streamwise direction on each side of the blade.

4.7.4. Specifying Inlet Turbulence Levels

It has been observed that the turbulence intensity specified at an inlet can decay quite rapidly depending on the inlet viscosity ratio (μ_t / μ) (and hence turbulence eddy frequency). As a result, the local turbulence intensity downstream of the inlet can be much smaller than the inlet value (see Figure 4.5: Exemplary Decay of Turbulence Intensity (Tu) as a Function of Streamwise Distance (x) (p. 85)). Typically, the larger the inlet viscosity ratio, the smaller the turbulent decay rate. However, if too large a viscosity ratio is specified (that is, > 100), the skin friction can deviate significantly from the laminar value. There is experimental evidence that suggests that this effect occurs physically; however, at this point it is not clear how accurately the transition model reproduces this behavior. For this reason, if possible, it is desirable to have a relatively low (that is, ≈1 – 10) inlet viscosity ratio and to estimate the inlet value of turbulence intensity such that at the leading edge of the blade/airfoil, the turbulence intensity has decayed to the desired value. The decay of turbulent kinetic energy can be calculated with the following analytical solution:

$$k=k_{inlet}\left(1+\omega_{inlet}\beta t\right)^{\frac{-\beta^*}{\beta}} \tag{4.191}$$

For the SST turbulence model in the freestream the constants are:

$$\beta^*=0.09,\beta=0.0828 \tag{4.192}$$

The time scale can be determined as follows:

$$t=\frac{x}{V} \tag{4.193}$$

where x is the streamwise distance downstream of the inlet and V is the mean convective velocity. The eddy viscosity is defined as:

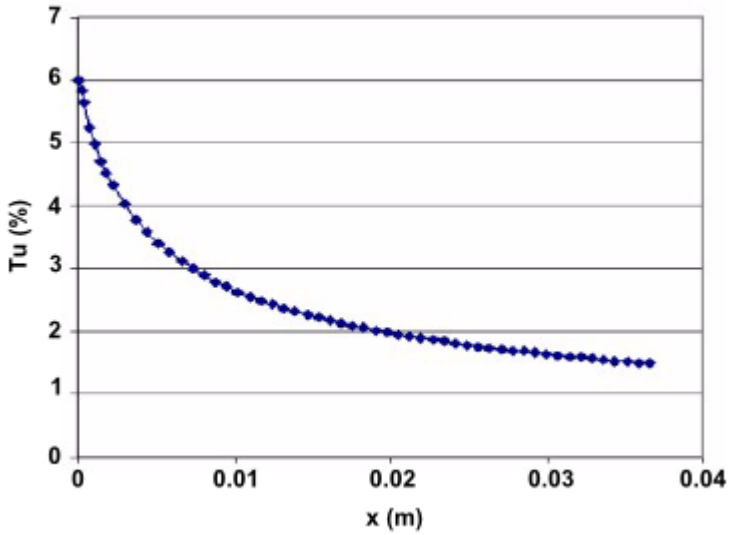
$$\mu_t=\frac{\rho k}{\omega} \tag{4.194}$$

The decay of turbulent kinetic energy equation can be rewritten in terms of inlet turbulence intensity (Tu_{inlet}) and inlet eddy viscosity ratio (μ_t / μ) as follows:

$$Tu=\left(Tu_{inlet}^2\left[1+\frac{3\rho V x\beta\left(Tu_{inlet}^2\right)}{2\mu\left(\mu_t/\mu\right)_{inlet}}\right]^{\frac{-\beta^*}{\beta}}\right)^{0.5} \tag{4.195}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Figure 4.5: Exemplary Decay of Turbulence Intensity (Tu) as a Function of Streamwise Distance (x)



You should ensure that the Tu values around the body of interest roughly satisfy Tu>0.1%. For smaller values of Tu, the reaction of the SST model production terms to the transition onset becomes too slow, and transition can be delayed past the physically correct location.

4.8. Intermittency Transition Model

This section describes the theory behind the Intermittency Transition model. Information is presented in the following sections:

4.8.1. Overview

4.8.2. Transport Equations for the Intermittency Transition Model

4.8.3. Coupling with the Other Models

4.8.4. Intermittency Transition Model and Rough Walls

For details about using the model in Ansys Fluent, see Modeling Turbulence and Setting Up the Algebraic or Intermittency Transition Model in the User's Guide.

4.8.1. Overview

The γ (intermittency) transition model is a further development based on the γ-Re_θ transition model (referred to as the Transition SST model in Ansys Fluent, as described in Transition SST Model (p. 77)). The γ transition model solves only one transport equation for the turbulence intermittency γ, and avoids the need for the second Re_θ equation of the γ-Re_θ transition model. The γ transition model has the following advantages over the γ-Re_θ transition model:

- It reduces the computational effort (by solving one transport equation instead of two).

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

- It avoids the dependency of the Re_θ equation on the velocity U. This makes the γ transition model Galilean invariant. It can therefore be applied to surfaces that move relative to the coordinate system for which the velocity field is computed.
- The model has provisions for crossflow instability that are not available for the k-kl or the γ-Re_θ transition model.
- The model formulation is simple and can be fine-tuned based on a small number of user parameters.

Like the γ-Re_θ transition model, the γ transition model is based strictly on local variables. The γ transition model is also only available in combination with the following turbulence models:

- BSL k-ω model
- SST k-ω model
- Scale-Adaptive Simulation with BSL or SST
- Detached Eddy Simulation with BSL or SST
- Shielded Detached Eddy Simulation (SDES) with BSL or SST
- Stress-Blended Eddy Simulation (SBES) with BSL or SST

Note the following limitations:

- The γ transition model is only applicable to wall-bounded flows. Like all other engineering transition models, the model is not applicable to transition in free shear flows. The model will predict free shear flows as fully turbulent.
- The γ transition model has only been calibrated for classical boundary layer flows. Application to other types of wall-bounded flows is possible, but might require a modification of the underlying correlations.
- The γ transition model has not been calibrated in combination with other physical effects that affect the source terms of the turbulence model, such as:
  - buoyancy
  - multiphase turbulence

4.8.2. Transport Equations for the Intermittency Transition Model

The transport equation for intermittency is the following:

$$\frac{\partial(\rho\gamma)}{\partial t}+\frac{\partial(\rho U_j\gamma)}{\partial x_j}=P_\gamma-E_\gamma+\frac{\partial}{\partial x_j}\left[\left(\mu+\frac{\mu_t}{\sigma_\gamma}\right)\frac{\partial\gamma}{\partial x_j}\right]$$

The formulation of the source terms is similar in nature to the corresponding terms in the γ-Re_θ transition model. The transition source term is defined as:

$$P_\gamma=F_{length}\rho S\gamma(1-\gamma)F_{onset}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

where S is the strain rate magnitude and F_{length}=100

The destruction/relaminarization source is defined as follows:

$$E_\gamma = c_{a2}\rho\Omega\gamma F_{turb}\left(c_{e2}\gamma - 1\right)$$

where Ω is the magnitude of the absolute vorticity rate, c_{a2}=0.06, c_{e2}=50 and $\Omega\gamma$=1.0. The transition onset is controlled by the following functions:

$$\begin{aligned} &F_{onset\,1} = \frac{\mathrm{Re}_V}{2.2\mathrm{Re}_{\theta c}}, \quad F_{onset\,2} = \min\left(F_{onset\,1}, 2.0\right) \\ &F_{onset\,3} = \max\left(1 - \left(\frac{R_T}{3.5}\right)^3, 0\right), \quad F_{onset} = \max\left(F_{onset\,2} - F_{onset\,3}, 0\right) \end{aligned} \tag{4.196}$$

$$F_{turb} = e^{-\left(\frac{R_T}{2}\right)^4}$$

$$R_T = \frac{\rho k}{\mu\omega}$$

$$\mathrm{Re}_V = \frac{\rho d_w^2 S}{\mu}$$

$$\mathrm{Re}_{\theta c} = f\left(Tu_L, \lambda_{\theta L}\right)$$

where d_w is the wall distance and $\mathrm{Re}_{\theta c}$ is a correlation used to trigger the transition model:

$$\mathrm{Re}_{\theta c}\left(Tu_L, \lambda_{\theta L}\right) = C_{TU1} + C_{TU2}\exp\left[-C_{TU3}\,Tu_L F_{PG}\left(\lambda_{\theta L}\right)\right]$$

In this correlation, $\mathrm{Re}_{\theta c}$ is the critical momentum thickness Reynolds number, and Tu_L and $\lambda_{\theta L}$ are locally defined variables that approximate the freestream turbulence intensity and the pressure gradient parameter, respectively. A local formulation that provides Tu_L levels in the middle of the boundary layer similar to the freestream turbulence intensity, Tu, is expressed as:

$$Tu_L = \min\left(100\frac{\sqrt{2k/3}}{\omega\, d_w}, 100\right)$$

The formula for $\lambda_{\theta L}$ reads as:

$$\lambda_{\theta L} = -0.1111\frac{dV}{dy}\frac{d_w^2}{\nu} + 0.1875 \tag{4.197}$$

For numerical robustness, $\lambda_{\theta L}$ was bounded as follows:

$$\lambda_{\theta L} = \min\left(\max\left(\lambda_{\theta L}, -10.0\right), 10.0\right) \tag{4.198}$$

The function F_{PG} accounts for the influence of the pressure gradient on transition, and is defined as:

$$F_{PG}\left(\lambda_\theta\right) = \begin{cases} \min\left(1 + C_{PG1}\lambda_\theta, C_{PG1}^{\lim}\right), & \lambda_\theta \geq 0 \\ \min\left(1 + C_{PG2}\lambda_\theta + C_{PG3}\min\left[\lambda_\theta + 1.0, 0\right], C_{PG2}^{\lim}\right), & \lambda_\theta < 0 \end{cases}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

The γ transition model has been calibrated against a wide range of generic as well as turbomachinery and external aeronautical test cases. While you should not have to adjust the model coefficients in most situations, the model provides access to the coefficients so that you can fine-tune the model. This should only be done based on detailed experimental data. You can adjust the following constants in the correlations: C_{TU1}, C_{TU2}, C_{TU3}, C_{PG1}, C_{PG2}, C_{PG3}, $C_{PG1}^{\lim}$, $C_{PG2}^{\lim}$, C_{SEP}, C_{RSF}.

The C_{TU1} constant defines the minimal value of critical $\mathrm{Re}_{\theta c}$ number, whereas the sum of C_{TU1} and C_{TU2} defines the maximal value of $\mathrm{Re}_{\theta c}$. The C_{TU3} controls how fast the $\mathrm{Re}_{\theta c}$ decreases as the turbulence intensity (Tu) increases. The C_{PG1} and C_{PG2} constants adjust the value of critical $\mathrm{Re}_{\theta c}$ number in areas with favorable and adverse pressure gradient, respectively. The C_{PG3} constant becomes active in regions with separation, correcting the $\mathrm{Re}_{\theta c}$ value if necessary. Additionally, the size of the separation bubble can be tuned by another constant, C_{SEP}, which enters the model through the source terms directly rather than the correlations. The constants $C_{PG1}^{\lim}$ and $C_{PG2}^{\lim}$ define the limits of the F_{PG} function; while they typically do not need to be adjusted, it is possible to do so (for details, contact your support engineer).

For crossflow transition, a widely used experimental correlation is the C1 correlation proposed by Arnal (1984) [223]. According to that correlation, transition occurs when the following condition is met:

$$\frac{\mathrm{Re}_{\delta 2t}^{*}}{150 f(H_S)} \geq 1$$

where

$$f(H_S) = \begin{cases} 1 & \text{for } H_S < 2.3 \\ \frac{2}{\pi}\arctan\left(\frac{0.106}{(H_S - 2.3)^{2.05}}\right) & \text{for } 2.3 < H_S < 2.7 \end{cases}$$

The crossflow Reynolds number is defined as:

$$\mathrm{Re}_{\delta_{\delta 2}^{*}} = \frac{U_{S,\delta}\delta_{\delta 2}^{*}}{\nu}$$

$$\delta_{\delta 2}^{*} = \int_0^{\infty} \frac{W_S}{U_{S,\delta}} dy$$

$$\delta_{XS}^{*} = \int_0^{\infty} \left(1 - \left(\frac{U_S}{U_{S,\delta}}\right)\right) dy$$

$$\Theta_{XS} = \int_0^{\infty} \left(1 - \frac{U_S}{U_{S,\delta}}\right) \frac{U_S}{U_{S,\delta}} dy$$

$$H_S = \frac{\delta_{XS}^{*}}{\Theta_{XS}}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

In the previous equations, $\mathrm{Re}_{\delta^*_{\delta 2}}$ is the crossflow Reynolds number, the index S refers to a coordinate system that is aligned with the freestream velocity $U_{S,\delta}$ (where δ refers to the boundary layer edge), W_S is the crossflow velocity normal to the direction of the freestream velocity. The value of 150 is the calibration constant of the Arnal C1 criterion corresponding to a critical Reynolds number.

The γ transition model accounts for the crossflow transition through a separate correlation, which approximates the C1 correlation through a local formulation. The formulation uses only local information and has the following functional form:

$$T_{C1\mathrm{local}}=\frac{C_{RSF}}{150}\left(G\Psi\mathrm{Re}_V\right)>1$$

The function G accounts for the influence of pressure gradient (shape factor); the function Ψ accounts for the ratio of the crossflow strength relative to the streamwise strength; the Reynolds number effect is included through Re_V. The correlation can be adjusted by the constant C_{RSF}. The transition location due to the crossflow instability will move upstream as C_{RSF} increases.

The function G is computed as follows:

$$g(\lambda_{\theta L,CF})=8.8\lambda^3_{\theta L,CF}-9.1\,\lambda^2_{\theta L,CF}+3.7\lambda_{\theta L,CF}+1.0$$
$$g(\lambda_{\theta L,CF})=\min\left[\max\left(g(\lambda_{\theta L,CF})\,,1.0\right),2.3\right]$$
$$G(\lambda_{\theta L,CF})=0.684/g(\lambda_{\theta L,CF})$$

where $\lambda_{\theta L,CF}$ is a local variable used to approximate the pressure gradient parameter and is defined similarly to $\lambda_{\theta L}$ (see Equation 4.197 (p. 87) and Equation 4.198 (p. 87)):

$$\lambda_{\theta L,CF}=-0.1111\frac{dV}{dy}\frac{d_{w2}}{\nu}+0.2555$$
$$\lambda_{\theta L,CF}=\min\left[\max\left(\lambda_{\theta L,CF},0\right),\,0.7\right]$$

The crossflow indicator function Ψ is a non-dimensional measure of the local crossflow strength relative to the streamwise strength. It is computed based on the wall normal change of the normalized vorticity vector $\vec{e}_\omega$, which serves as a measure of the three-dimensionality of the velocity profile and which becomes zero for 2D flow.

$$\vec{e}_\omega=\frac{\vec{\omega}}{|\vec{\omega}|}$$

$$\vec{\phi}=\vec{n}\cdot\nabla\vec{e}_\omega$$

$$\Psi=|\vec{\phi}|\,d_w$$

Here, $\vec{\omega}$ is the vorticity vector and $\vec{n}$ is the wall normal vector.

The indicator function Ψ is proportional to the local change of the flow angle and therefore a measure of the overall crossflow strength.

The triggering function that accounts for the crossflow effects is defined as follows:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$F_{\text{onset},CF}=\min[\max[\ 100\,(T_{C1local}-1),0],1]\ .$$

The source term in the γ equation is then triggered, either by the primary triggering function F_{onset} (see Equation 4.196 (p. 87)) of the Intermittency transition model, or by $F_{\text{onset},CF}$: whichever has a maximal value.

$$F_{\text{onset}}=\max[F_{\text{onset}}\,,F_{\text{onset},CF}]$$

The factor of 100 is used in $F_{\text{onset},CF}$ in order to speed up the transition process.

The default values for the model constants are the following:

C_{TU1} = 100.0, C_{TU2} = 1000.0, C_{TU3} = 1.0, C_{PG1} = 1.0, C_{PG2} = -0.5, C_{PG3} = 0.0, $C_{PG1}^{\lim}$ = 1.5, $C_{PG2}^{\lim}$ = 3.0, C_{SEP} = 1.0, C_{RSF} = 1.0

4.8.3. Coupling with the Other Models

The coupling between the γ transition model and the BSL k-ω, SST k-ω, SAS with BSL or SST, and DES with BSL or SST models is similar as for the γ-Re_θ transition model (see Coupling the Transition Model and SST Transport Equations (p. 80)).

4.8.4. Intermittency Transition Model and Rough Walls

Note that the correlation does not include the effects of rough walls. If you plan to include rough walls, it is recommended that you switch to the γ-Re_θ transition model (that is, the Transition SST model).

4.9. Algebraic Transition Model

This section describes the theory behind the Algebraic Transition model. Information is presented in the following sections:

4.9.1. Overview

4.9.2. Model Formulation

4.9.3. Transition Correlation

4.9.4. Pressure Gradient Function

4.9.5. Bubble Separation

4.9.6. Coefficients for the Algebraic Transition Model

4.9.7. Algebraic Transition Model and Rough Walls

For details about using the model in Ansys Fluent, see Setting Up the Algebraic or Intermittency Transition Model in the *Fluent User's Guide*.

4.9.1. Overview

The algebraic transition model belongs to the family of Local-Correlation-based Transition Models (LCTM) and is a further simplification of the Intermittency Transition Model which is based on a single

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

transport equation for the intermittency γ (gamma-transport-eqn). The simplification avoids using the transport equation and formulates the intermittency as an algebraic equation. This is possible because it has been observed that in the transport equation-based models, the source terms are dominant compared to the convection/diffusion terms.

4.9.2. Model Formulation

The intermittency multiplies the production term P_k in the equation for the turbulent kinetic energy:

$$P_k \rightarrow \gamma P_k$$

This is the only modification to the underlying turbulence model. The new transition model is compatible with k-ω based turbulence models.

The intermittency is computed from an algebraic formula. The full model formulation will be published shortly and only the essential elements of the formulation are given here. The underlying concept is that transition is triggered by a correlation which contains the major physical effects. The critical displacement thickness Reynolds number is termed $\mathrm{Re}_{\theta C}$ and contains the correlation which fits the experimental transition locations. $\mathrm{Re}_{\theta C}$ is a function of the turbulence intensity Tu and the non-dimensional pressure gradient parameter λ_θ. Both are computed locally in the same way as in the existing one-equation γ-model (Intermittency Transition Model in the *Fluent Theory Guide* (p. 85)):

$$Tu_l = \min\left(100\frac{\sqrt{2k/3}}{\omega d_w}, 100\right) \tag{4.199}$$

$$\lambda_{\theta l} = -0.1111\frac{dV}{dy}\frac{y^2}{\nu} + 0.1875 \tag{4.200}$$

For numerical robustness, $\lambda_{\theta l}$ is bounded as follows:

$$\lambda_{\theta l} = C_{\lambda,scale}\min\left(\max\left(\lambda_{\theta l}, -10.0, 10.0\right)\right) \tag{4.201}$$

As inherent to the LCTM concept, the formulation is based on the ratio between the actual state of the boundary layer described by the vorticity Reynolds number Re_{V22} and the critical Reynolds number $\mathrm{Re}_{\theta C}$:

$$Rat_{\mathrm{Re}} = \frac{\mathrm{Re}_{V22}}{\mathrm{Re}_{\theta C}} \tag{4.202}$$

$$\mathrm{Re}_{V22} = \min\left(\frac{\widetilde{\Omega} d_w^2}{2.2\nu}, 5000\right) \tag{4.203}$$

$$\mathrm{Re}_{\theta C} = g\left(Tu_l, \lambda_{\theta l}\right) \tag{4.204}$$

The intermittency is then computed from:

$$\gamma = \tanh\left(Rat_{\mathrm{Re}}^2\right) \tag{4.205}$$

The quantity $\widetilde{\Omega}$ is defined as $\widetilde{\Omega} = \max\left(S, \Omega, 0.1\omega\right)$ with the strain rate S, the vorticity rate Ω, and the ω from the ω-equation of the underlying turbulence model.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

4.9.3. Transition Correlation

The key to the definition of the transition location lies in the formulation of the transition onset criterion $\mathrm{Re}_{\theta C}$. To explain the concept, the correlation generically has the form:

$$\mathrm{Re}_{\theta C}=\min(\frac{210}{Tu_l},1000)FPG(\lambda_{\theta l}) \tag{4.206}$$

when using the default constants, which is essentially a $1/Tu_l$ relation with limiter. The function FPG contains the dependence on the pressure gradient $\lambda_{\theta l}$ which will be discussed later.

In more detail, the correlation reads:

$$\mathrm{Re}_{\theta C}=\min(\frac{\mathrm{Re}C_{C1}}{Tu_l^{\mathrm{Re}C_{C2}}}CTuCrit,\mathrm{Re}C_{Max})FPG(\lambda_{\theta l}) \tag{4.207}$$

The quantities $\mathrm{Re}C_{C1}$, $\mathrm{Re}C_{C2}$, and $\mathrm{Re}C_{Max}$ are constants and $CTuCrit$ is a blended coefficient (blending between low and high Re number situations):

$$CTuCrit=CTu_{High}(1-FRV1)+CTu_{Low}FRV1 \tag{4.208}$$

with CTu_{High} and CTu_{Low} being constants. The blending function between low and high Re number transition reads:

$$FRV1=\min\left[\left(\frac{\mathrm{Re}_{V22}}{\mathrm{RV1}_{Switch}}\right)^2,1\right] \tag{4.209}$$

with $\mathrm{RV1}_{Switch}$ being a constant. This function is 0 for low and 1 for high Re numbers. The purpose of this function is to allow for a different calibration for the different Re number regimes (with low Re typically being bypass transition and high Re number being natural transition).

4.9.4. Pressure Gradient Function

The formulation for FPG is different for favorable and adverse pressure gradients:

$$FPG=\min\left[\max(1.0-CFPG\cdot\max(\lambda_{\theta l},0.0),0.25),4.0\right] if\ \lambda_{\theta l}\geq 0 \tag{4.210}$$

$$FPG=\min\left[\max(1.0-CAPG\cdot\min(\lambda_{\theta l},0.0),0.25),4.0\right] if\ \lambda_{\theta l}<0 \tag{4.211}$$

The impact in either situation on the basic correlation is therefore limited by a factor 4. The coefficients *CFPG* and *CAPG* are again blended for low and high Re numbers:

$$CFPG=CFPG_{High}(1-FRV1)+CFPG_{Low}FRV1 \tag{4.212}$$

$$CAPG=CAPG_{High}(1-FRV1)+CAPG_{Low}FRV1 \tag{4.213}$$

4.9.5. Bubble Separation

This transition mechanism occurs in cases where the laminar boundary layer separates and transition takes place inside the bubble. The size of the bubble depends on many factors, like the Re number, the pressure gradient and the turbulence intensity.

The current formulation enhances the intermittency γ to higher values than γ=1 in the bubble regime, as it was observed that otherwise the turbulence models might not react quickly enough to close the bubble under low Tu conditions.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$\gamma \rightarrow \gamma + \gamma_{bubble} \tag{4.214}$$

$$\gamma_{bubble} = \min(CBUB_{C1} \cdot \gamma \cdot \phi_1 \cdot \phi_2 \cdot \phi_3, 3.0) \tag{4.215}$$

$$\phi_1 = \frac{\max(\mu - \frac{1}{2}\mu_t, 0)}{0.2\mu + \mu_t} \tag{4.216}$$

$$\phi_2 = \min\left[\left(\frac{\mathrm{Re}_{V22H}}{300}\right)^2, 1\right] \tag{4.217}$$

where

$$\mathrm{Re}_{V22H} = \frac{\min(\mathrm{Re}_{V22}, 1000.0) * \min(S, \Omega)}{\max(\widetilde{\Omega}, 1.0e^{-9})} \tag{4.218}$$

$$\phi_3 = \min\left[\max(Rat_{\mathrm{Re}} - CBUB_{C2}, 0), 5\right] \tag{4.219}$$

4.9.6. Coefficients for the Algebraic Transition Model

The coefficients are (defaults in parenthesis):

- $C_{APG,hightu}$(-0.5) - Parameter ($CAPG_{High}$) to calibrate impact of adverse pressure gradients for high Tu regimes.
- $C_{APG,lowtu}$(-0.5) - Parameter ($CAPG_{Low}$) to calibrate impact of adverse pressure gradients for low Tu regimes.
- $C_{FPG,hightu}$(0.6) - Parameter ($CFPG_{High}$) to calibrate impact of favorable pressure gradients for high Tu regimes.
- $C_{FPG,lowtu}$(1.0) - Parameter ($CFPG_{Low}$) to calibrate impact of favorable pressure gradients for low Tu regimes.
- $C_{\lambda,scale}$(1.0) -Parameter to scale the non-dimensional pressure gradient parameter λ_θ.
- $C_{TU,hightu}$(1.06) - Parameter (CTu_{High}) to calibrate impact of Tu for high Tu regimes.
- $C_{TU,lowtu}$(2.25) - Parameter (CTu_{Low}) to calibrate impact of Tu for low Tu regimes.
- Re_C(1.0) - Critical Reynolds number correlation; built-in version or user-defined via UDF.
- $\mathrm{Re}_{C,c1}$(210) - Parameter ($\mathrm{Re}C_{C1}$) in correlation.
- $\mathrm{Re}_{C,c2}$(1.0) - Parameter ($\mathrm{Re}C_{C2}$) in correlation.
- $\mathrm{Re}_{C,\max}$(1000.0) - Parameter ($\mathrm{Re}C_{Max}$) in correlation - specifies upper limit on transition location as Tu goes to zero.
- $RV1_{switch}$(1000.0) - Parameter ($RV1_{switch}$) - steers the switching function between high and low Tu levels.
- $C_{bubble,c1}$(0.5) - Parameter ($CBUB_{C1}$) - to steer bubble modification.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

- $C_{bubble,c2}$(2.5) - Parameter ($CBUB_{C2}$) - to steer bubble modification.

4.9.7. Algebraic Transition Model and Rough Walls

Note that the algebraic transition model does not include the effects of rough walls. If you plan to include rough walls, it is recommended that you switch to the γ -Re_theta transition model (that is, the Transition SST model).

4.10. Explicit Algebraic Reynolds Stress Model (EARSM)

Explicit Algebraic Reynolds Stress Models (EARSM) represent an extension of the standard two-equation models. They are derived from the Reynolds stress transport equations and give a nonlinear relation between the Reynolds stresses and the mean strain-rate and vorticity tensors. Due to the higher order terms, many flow phenomena are included in the model without the need to solve transport equations for individual Reynolds stresses. The WJ-BSL-EARSM allows an extension of the baseline (BSL) $k-\omega$ turbulence model to capture the following flow effects:

- Anisotropy of Reynolds stresses
- Secondary flows

For information on enabling the WJ-BSL-EARSM model, see Setting up the WJ-BSL-EARSM Model in the *Fluent User's Guide*.

The BSL model blends the robust and accurate formulation of the k-ω model in the near-wall region with the freestream independence of the k-ε model in the far field. The BSL model was developed by Menter [425] (p. 1077) and is described in Baseline (BSL) k-ω Model (p. 65)). There is also an option **GEKO** which allows the combination of the EARSM with the Generalized $k-\omega$ (GEKO) Model (see Generalized k-ω (GEKO) Model (p. 70) for more information on the GEKO model).

The implementation of the EARSM in Ansys Fluent is based on the explicit algebraic Reynolds stress model of Wallin and Johansson [680] (p. 1092). Differences from the original formulation by Wallin and Johansson are explained in the following text.

With EARSM, the Reynolds stresses are computed from the anisotropy tensor according to its definition:

$$\overline{u_i u_j} = k\left(a_{ij} + \frac{2}{3}\delta_{ij}\right)$$

where the anisotropy tensor a_{ij} is searched as a solution of the following implicit algebraic matrix equation:

$$N\mathbf{a} = -A_1\mathbf{S} + \left(\mathbf{a\Omega} - \mathbf{\Omega a}\right) - A_2\left(\mathbf{aS} - \mathbf{Sa} - \frac{2}{3}tr\{\mathbf{aS}\}\right), \text{ with } \quad N = A_3 + A_4\left(P_k/\varepsilon\right) \tag{4.220}$$

The coefficients A_i in this matrix equation depend on the C_i-coefficients of the pressure-strain term in the underlying Reynolds stress transport model. Their values are selected here as A_1=1.245, A_2=0, A_3=1.8, A_4=2.25.

The values of A_2, A_3, and A_4 are the same as those used in the original model by Wallin and Johansson [680] (p. 1092). As for the value of A_1, it is increased from 1.2 to 1.245 in the course of calibrating EARSM for its use together with the BSL $k-\omega$ model.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

S_{ij} and Ω_{ij} denote the non-dimensional strain-rate and vorticity tensors, respectively. They are defined as:

$$S_{ij}=\frac{1}{2}\tau\left(\frac{\partial U_i}{\partial x_j}+\frac{\partial U_j}{\partial x_i}\right) \tag{4.221}$$

$$\Omega_{ij}=\frac{1}{2}\tau\left(\frac{\partial U_i}{\partial x_j}-\frac{\partial U_j}{\partial x_i}\right) \tag{4.222}$$

where the time-scale τ is given by:

$$\tau=k/\varepsilon=1/\left(C_\mu\omega\right), \quad C_\mu=0.09 \tag{4.223}$$

In order to arrive at an explicit solution of the Equation 4.220 (p. 94), the anisotropy tensor is expressed as a polynomial based on the strain rate and the vorticity tensors as follows:

$$\begin{aligned} a_{ij}=&\beta_1 S_{ij}+\beta_3\left(\Omega_{ik}\Omega_{kj}-\tfrac{1}{3}II_\Omega\delta_{ij}\right)+\beta_4\left(S_{ik}\Omega_{kj}-\Omega_{ik}S_{kj}\right)\\ &+\beta_6\left(S_{ik}\Omega_{kl}\Omega_{lj}+\Omega_{ik}\Omega_{kl}S_{lj}-\tfrac{2}{3}IV\delta_{ij}-II_\Omega S_{ij}\right)\end{aligned} \tag{4.224}$$

The β-coefficients are evaluated to:

$$\beta_1=-N/Q$$

$$\beta_3=-12\cdot IV/\left(N\cdot Q\cdot\left(2N^2-II_\Omega\right)\right)$$

$$\beta_4=-1/Q$$

$$\beta_6=-6\cdot N/\left(Q\cdot\left(2N^2-II_\Omega\right)\right)$$

where the denominator Q is:

$$Q=\left(N^2-2II_\Omega\right)/A_1$$

The invariants, which appear in the formulation of the anisotropy tensor and the coefficients, are defined by:

$$II_S=S_{kl}S_{lk}$$

$$II_\Omega=\Omega_{kl}\Omega_{lk}$$

$$IV=S_{kl}\Omega_{lm}\Omega_{mk}$$

The model representation of the anisotropy tensor Equation 4.224 (p. 95) and its coefficients β_i follows the original model by Wallin and Johansson [680] (p. 1092) with two differences. First, the fourth order tensor polynomial contribution (the $\beta_9\left(\mathbf{\Omega S\Omega}^2-\mathbf{\Omega}^2\mathbf{S\Omega}\right)$ term) is neglected in Equation 4.224 (p. 95). Second, the tensor basis is slightly changed for convenience according to Apsley and Leschziner [23] (p. 1054). Although the tensor basis is changed, the model remains algebraically equivalent to the

original model of Wallin and Johansson. The latter change results in correspondingly changed expressions for the coefficients β_i.

In three-dimensional flows, the equation to be solved for the function N is of sixth order and no explicit solution can be derived, whereas in two-dimensional mean flows the function N can be derived from a cubic equation, an analytic solution of which is recommended by Wallin and Johansson [680] (p. 1092) also for three-dimensional cases:

$$N=\begin{cases} A_3/3+\left(P_1+\sqrt{P_2}\right)^{1/3}+\text{sign}\left(P_1-\sqrt{P_2}\right)\left|P_1-\sqrt{P_2}\right|^{1/3} & \text{for } P_2\geq 0 \\ A_3/3+2\left(P_1^2-P_2\right)^{1/6}\cos\left(\frac{1}{3}\arccos\left(\frac{P_1}{\sqrt{P_1^2-P_2}}\right)\right) & \text{for } P_2<0 \end{cases} \tag{4.225}$$

where

$$P_1=\left(\frac{A_3^2}{27}+\frac{A_1A_4}{6}II_S-\frac{2}{3}II_\Omega\right)\cdot A_3$$

$$P_2=P_1^2-\left(\frac{A_3^2}{9}+\frac{A_1A_4}{3}II_S+\frac{2}{3}II_\Omega\right)^3$$

In the original model by Wallin and Johansson [680] (p. 1092), the diffusion terms in the transport equations for k and ω were calculated using the effective eddy viscosity, $\mu_t^{eff}=\left(C_\mu^{eff}/C_\mu\right)\cdot k/\omega$, of EARSM, where $C_\mu^{eff}=-\beta_1/2$. The EARSM model, implemented in Ansys Fluent, uses the standard eddy viscosity $\mu_t=k/\omega$ for the diffusion terms. This model change helps to avoid the problems with the asymptotic behavior at the boundary layer edge, which were reported by Hellsten [236] (p. 1066).

When in combination with the BSL model, the standard coefficients of the underlying BSL k-ω model are used. When in combination with the GEKO model, then the EARSM model uses the GEKO model coefficients.

4.11. Reynolds Stress Model (RSM)

This section describes the theory behind the Reynolds stress model (RSM). Information is presented in the following sections:

4.11.1. Overview

4.11.2. Reynolds Stress Transport Equations

4.11.3. Modeling Turbulent Diffusive Transport

4.11.4. Modeling the Pressure-Strain Term

4.11.5. Effects of Buoyancy on Turbulence

4.11.6. Modeling the Turbulence Kinetic Energy

4.11.7. Modeling the Dissipation Rate

4.11.8. Modeling the Turbulent Viscosity

4.11.9. Wall Boundary Conditions

4.11.10. Convective Heat and Mass Transfer Modeling

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

For details about using the model in Ansys Fluent, see Modeling Turbulence and Setting Up the Reynolds Stress Model in the User's Guide.

4.11.1. Overview

The Reynolds stress model (RSM) [203] (p. 1064), [336] (p. 1072), [337] (p. 1072) is the most elaborate type of RANS turbulence model that Ansys Fluent provides. Abandoning the isotropic eddy-viscosity hypothesis, the RSM closes the Reynolds-averaged Navier-Stokes equations by solving transport equations for the Reynolds stresses, together with an equation for the dissipation rate. This means that five additional transport equations are required in 2D flows, in comparison to seven additional transport equations solved in 3D.

Since the RSM accounts for the effects of streamline curvature, swirl, rotation, and rapid changes in strain rate in a more rigorous manner than one-equation and two-equation models, it has greater potential to give accurate predictions for complex flows. However, the fidelity of RSM predictions is still limited by the closure assumptions employed to model various terms in the exact transport equations for the Reynolds stresses. The modeling of the pressure-strain and dissipation-rate terms is particularly challenging, and often considered to be responsible for compromising the accuracy of RSM predictions.

In addition, even RSM rely on scale equations (ε- or ω- / BSL-) and inherit deficiencies resulting from the underlying assumptions in these equations. Ansys Fluent offers the following model combinations:

- ε-based Reynolds stress models:
  - linear pressure-strain model
  - quadratic pressure-strain model
- ω-based Reynolds stress models:
  - stress-omega model
  - stress-BSL model

The default Reynolds stress model in Ansys Fluent is based on the ε-equation, and uses a linear model for the pressure-strain term. The second ε-based model allows the usage of a quadratic model for the pressure-strain term.

The two Reynolds stress models based on the ω-equation both use a linear model for the pressure-strain term, but differ with regard to the scale equation: the stress-omega model is based on the ω-equation, whereas the stress-BSL model solves the scale equation from the baseline (BSL) k-ω model and thus removes the free-stream sensitivity observed with the stress-omega model.

The RSM might not always yield results that are clearly superior to the simpler models in all classes of flows to warrant the additional computational expense. However, use of the RSM is a must when the flow features of interest are the result of anisotropy in the Reynolds stresses. Among the examples are cyclone flows, highly swirling flows in combustors, rotating flow passages, and the stress-induced secondary flows in ducts.

The exact form of the Reynolds stress transport equations may be derived by taking moments of the exact momentum equation. This is a process wherein the exact momentum equations for the fluctuations are multiplied by the fluctuating velocities and averaged, the product then being Reynolds-

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

averaged. Unfortunately, several of the terms in the exact equation are unknown and modeling assumptions are required in order to close the equations.

4.11.2. Reynolds Stress Transport Equations

The exact transport equations for the transport of the Reynolds stresses, $\rho\overline{u_i'u_j'}$, may be written as follows:

$$\underbrace{\frac{\partial}{\partial t}\left(\rho\overline{u_i'u_j'}\right)}_{\text{Local Time Derivative}} + \underbrace{\frac{\partial}{\partial x_k}\left(\rho u_k\overline{u_i'u_j'}\right)}_{C_{ij}\equiv\text{Convection}} = \underbrace{-\frac{\partial}{\partial x_k}\left[\rho\overline{u_i'u_j'u_k'}+\overline{p'\left(\delta_{kj}u_i'+\delta_{ik}u_j'\right)}\right]}_{D_{T,ij}\equiv\text{Turbulent Diffusion}} + \underbrace{\frac{\partial}{\partial x_k}\left[\mu\frac{\partial}{\partial x_k}\left(\overline{u_i'u_j'}\right)\right]}_{D_{L,ij}\equiv\text{Molecular Diffusion}} \underbrace{-\rho\left(\overline{u_i'u_k'}\frac{\partial u_j}{\partial x_k}+\overline{u_j'u_k'}\frac{\partial u_i}{\partial x_k}\right)}_{P_{ij}\equiv\text{Stress Production}} \underbrace{-\rho\beta\left(g_i\overline{u_j'\theta}+g_j\overline{u_i'\theta}\right)}_{G_{ij}\equiv\text{Buoyancy Production}} + \underbrace{\overline{p'\left(\frac{\partial u_i'}{\partial x_j}+\frac{\partial u_j'}{\partial x_i}\right)}}_{\varphi_{ij}\equiv\text{Pressure Strain}} \underbrace{-2\mu\overline{\frac{\partial u_i'}{\partial x_k}\frac{\partial u_j'}{\partial x_k}}}_{\varepsilon_{ij}\equiv\text{Dissipation}} \underbrace{-2\rho\Omega_k\left(\overline{u_j'u_m'}\varepsilon_{ikm}+\overline{u_i'u_m'}\varepsilon_{jkm}\right)}_{F_{ij}\equiv\text{Production by System Rotation}} + \underbrace{S_{user}}_{\text{User-Defined Source Term}} \tag{4.226}$$

Of the various terms in these exact equations, C_{ij}, $D_{L,ij}$, P_{ij}, and F_{ij} do not require any modeling. However, $D_{T,ij}$, G_{ij}, φ_{ij}, and ε_{ij} need to be modeled to close the equations. The following sections describe the modeling assumptions required to close the equation set.

4.11.3. Modeling Turbulent Diffusive Transport

For the ε-based RSM, $D_{T,ij}$ can be modeled by the generalized gradient-diffusion model of Daly and Harlow [132] (p. 1060):

$$D_{T,ij}=C_s\frac{\partial}{\partial x_k}\left(\rho\frac{k\overline{u_k'u_\ell'}}{\varepsilon}\frac{\partial\overline{u_i'u_j'}}{\partial x_\ell}\right) \tag{4.227}$$

However, this equation can result in numerical instabilities, so it has been simplified in Ansys Fluent to use a scalar turbulent diffusivity as follows [368] (p. 1074):

$$D_{T,ij}=\frac{\partial}{\partial x_k}\left(\frac{\mu_t}{\sigma_k}\frac{\partial\overline{u_i'u_j'}}{\partial x_k}\right) \tag{4.228}$$

The turbulent viscosity, μ_t, is computed using Equation 4.259 (p. 104) for the ε-based RSM.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Lien and Leschziner [368] (p. 1074) derived a value of σ_k=0.82 by applying the generalized gradient-diffusion model, Equation 4.227 (p. 98), to the case of a planar homogeneous shear flow. Note that this value of σ_k is different from that in the standard and realizable k-ε models, in which σ_k=1.0.

For the ω-based Reynolds stress models, Equation 4.228 (p. 98) is also used to model the turbulent diffusive transport, and the turbulent viscosity is calculated from Equation 4.74 (p. 62).

The stress-omega model sets σ_k to a value of 2.0, whereas the stress-BSL model blends the value between the inner region ($\sigma_{k,1}$=2.0) and outer region ($\sigma_{k,2}$=1.0) using the blending function F_1 from Equation 4.107 (p. 66):

$$\sigma_k = F_1\sigma_{k,1} + \left(1 - F_1\right)\sigma_{k,2} \tag{4.229}$$

4.11.4. Modeling the Pressure-Strain Term

Ansys Fluent offers the following Reynolds stress models based on the ε- or ω / BSL-equation:

- ε-based Reynolds stress models:
  - linear pressure-strain model
  - quadratic pressure-strain model
- ω-based Reynolds stress models:
  - stress-omega model
  - stress-BSL model

4.11.4.1. Linear Pressure-Strain Model

By default in Ansys Fluent, the pressure-strain term, φ_{ij}, in Equation 4.226 (p. 98) is modeled according to the proposals by Gibson and Launder [203] (p. 1064), Fu et al. [194] (p. 1064), and Launder [335] (p. 1072), [336] (p. 1072).

The classical approach to modeling φ_{ij} uses the following decomposition:

$$\varphi_{ij} = \varphi_{ij,1} + \varphi_{ij,2} + \varphi_{ij,w} \tag{4.230}$$

where $\varphi_{ij,1}$ is the slow pressure-strain term, also known as the return-to-isotropy term, $\varphi_{ij,2}$ is called the rapid pressure-strain term, and $\varphi_{ij,w}$ is the wall-reflection term.

The slow pressure-strain term, $\varphi_{ij,1}$, is modeled as

$$\varphi_{ij,1} \equiv -C_1\rho\frac{\varepsilon}{k}\left[\overline{u_i'u_j'} - \frac{2}{3}\delta_{ij}k\right] \tag{4.231}$$

with C_1 = 1.8.

The rapid pressure-strain term, $\varphi_{ij,2}$, is modeled as

$$\varphi_{ij,2} \equiv -C_2\left[\left(P_{ij} + F_{ij} + 5/\,6G_{ij} - C_{ij}\right) - \frac{2}{3}\delta_{ij}\left(P + 5/\,6G - C\right)\right] \tag{4.232}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

where C_2 = 0.60, P_{ij}, F_{ij}, G_{ij}, and C_{ij} are defined as in Equation 4.226 (p. 98), $P=\frac{1}{2}P_{kk}$, $G=\frac{1}{2}G_{kk}$, and $C=\frac{1}{2}C_{kk}$.

The wall-reflection term, $\varphi_{ij,w}$, is responsible for the redistribution of normal stresses near the wall. It tends to damp the normal stress perpendicular to the wall, while enhancing the stresses parallel to the wall. This term is modeled as

$$\begin{aligned}\varphi_{ij,w} \equiv\ & C_1'\frac{\varepsilon}{k}\left(\overline{u_k'u_m'}n_kn_m\delta_{ij}-\frac{3}{2}\overline{u_i'u_k'}n_jn_k-\frac{3}{2}\overline{u_j'u_k'}n_in_k\right)\frac{C_\ell k^{3/2}}{\varepsilon d}\\ &+C_2'\left(\varphi_{km,2}n_kn_m\delta_{ij}-\frac{3}{2}\varphi_{ik,2}n_jn_k-\frac{3}{2}\varphi_{jk,2}n_in_k\right)\frac{C_\ell k^{3/2}}{\varepsilon d}\end{aligned} \tag{4.233}$$

where $C_1'=0.5$, $C_2'=0.3$, n_k is the x_k component of the unit normal to the wall, d is the normal distance to the wall, and $C_\ell=C_\mu^{3/4}/\kappa$, where $C_\mu=0.09$ and κ is the von Kármán constant (= 0.4187).

$\varphi_{ij,w}$ is included by default in the Reynolds stress model.

4.11.4.2. Low-Re Modifications to the Linear Pressure-Strain Model

When the RSM is applied to near-wall flows using the enhanced wall treatment described in Two-Layer Model for Enhanced Wall Treatment (p. 145), the pressure-strain model must be modified. The modification used in Ansys Fluent specifies the values of C_1, C_2, C_1', and C_2' as functions of the Reynolds stress invariants and the turbulent Reynolds number, according to the suggestion of Launder and Shima [339] (p. 1072):

$$C_1 = 1+2.58AA_2^{0.25}\left\{1-\exp\left[-\left(0.0067Re_t\right)^2\right]\right\} \tag{4.234}$$

$$C_2 = 0.75\sqrt{A} \tag{4.235}$$

$$C_1' = -\frac{2}{3}C_1+1.67 \tag{4.236}$$

$$C_2' = \max\left[\frac{\frac{2}{3}C_2-\frac{1}{6}}{C_2},0\right] \tag{4.237}$$

with the turbulent Reynolds number defined as $Re_t=\left(\rho k^2/\mu\varepsilon\right)$. The flatness parameter A and tensor invariants, A_2 and A_3, are defined as

$$A \equiv \left[1-\frac{9}{8}\left(A_2-A_3\right)\right] \tag{4.238}$$

$$A_2 \equiv a_{ik}a_{ki} \tag{4.239}$$

$$A_3 \equiv a_{ik}a_{kj}a_{ji} \tag{4.240}$$

a_{ij} is the Reynolds-stress anisotropy tensor, defined as

$$a_{ij}=-\left(\frac{-\rho\overline{u_i'u_j'}+\frac{2}{3}\rho k\delta_{ij}}{\rho k}\right) \tag{4.241}$$

The modifications detailed above are employed only when the enhanced wall treatment is selected in the Viscous Model Dialog Box.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

4.11.4.3. Quadratic Pressure-Strain Model

An optional pressure-strain model proposed by Speziale, Sarkar, and Gatski [622] (p. 1089) is provided in Ansys Fluent. This model has been demonstrated to give superior performance in a range of basic shear flows, including plane strain, rotating plane shear, and axisymmetric expansion/contraction. This improved accuracy should be beneficial for a wider class of complex engineering flows, particularly those with streamline curvature. The quadratic pressure-strain model can be selected as an option in the Viscous Model Dialog Box.

This model is written as follows:

$$\begin{aligned}\varphi_{ij} = & -\left(C_1\rho\varepsilon + C_1^*P\right)b_{ij} + C_2\rho\varepsilon\left(b_{ik}b_{kj} - \frac{1}{3}b_{mn}b_{mn}\delta_{ij}\right) + \left(C_3 - C_3^*\sqrt{b_{ij}b_{ij}}\right)\rho k S_{ij} \\ & + C_4\rho k\left(b_{ik}S_{jk} + b_{jk}S_{ik} - \frac{2}{3}b_{mn}S_{mn}\delta_{ij}\right) + C_5\rho k\left(b_{ik}\Omega_{jk} + b_{jk}\Omega_{ik}\right)\end{aligned} \tag{4.242}$$

where b_{ij} is the Reynolds-stress anisotropy tensor defined as

$$b_{ij} = -\left(\frac{-\rho\overline{u_i'u_j'} + \frac{2}{3}\rho k\delta_{ij}}{2\rho k}\right) \tag{4.243}$$

The mean strain rate, S_{ij}, is defined as

$$S_{ij} = \frac{1}{2}\left(\frac{\partial u_j}{\partial x_i} + \frac{\partial u_i}{\partial x_j}\right) \tag{4.244}$$

The mean rate-of-rotation tensor, Ω_{ij}, is defined by

$$\Omega_{ij} = \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} - \frac{\partial u_j}{\partial x_i}\right) \tag{4.245}$$

The constants are

$$C_1 = 3.4, C_1^* = 1.8, C_2 = 4.2, C_3 = 0.8, C_3^* = 1.3, C_4 = 1.25, C_5 = 0.4$$

The quadratic pressure-strain model does not require a correction to account for the wall-reflection effect in order to obtain a satisfactory solution in the logarithmic region of a turbulent boundary layer. It should be noted, however, that the quadratic pressure-strain model is not available when the enhanced wall treatment is selected in the Viscous Model Dialog Box.

4.11.4.4. Stress-Omega Model

The stress-omega model is a stress-transport model that is based on the omega equations and LRR model [705] (p. 1094). This model is ideal for modeling flows over curved surfaces and swirling flows. The stress-omega model can be selected in the Viscous Model Dialog Box and requires no treatments of wall reflections. The closure coefficients are identical to the k-ω model (Model Constants (p. 65)), however, there are additional closure coefficients, C_1 and C_2, noted below.

The stress-omega model resembles the k-ω model due to its excellent predictions for a wide range of turbulent flows. Furthermore, low Reynolds number modifications and surface boundary conditions for rough surfaces are similar to the k-ω model.

Equation 4.230 (p. 99) can be re-written for the stress-omega model such that wall reflections are excluded:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$\varphi_{ij}=\varphi_{ij,1}+\varphi_{ij,2} \tag{4.246}$$

Therefore,

$$\begin{aligned}\phi_{ij}= \quad & -C_1\rho\beta^*_{RSM}\omega\left[\overline{u'_iu'_j}-{}^2\!/\!_3\,\delta_{ij}k\right]-\hat{\alpha}_0\left[P_{ij}-{}^1\!/\!_3\,P_{kk}\delta_{ij}\right] \\ & -\hat{\beta}_0\left[D_{ij}-{}^1\!/\!_3\,P_{kk}\delta_{ij}\right]-k\hat{\gamma}_0\left[S_{ij}-{}^1\!/\!_3\,S_{kk}\delta_{ij}\right]\end{aligned} \tag{4.247}$$

where D_{ij} is defined as

$$D_{ij}=-\rho\left[\overline{u_i'u_m'}\frac{\partial u_m}{\partial x_j}+\overline{u_j'u_m'}\frac{\partial u_m}{\partial x_i}\right] \tag{4.248}$$

The mean strain rate S_{ij} is defined in Equation 4.244 (p. 101) and β^*_{RSM} is defined by

$$\beta^*_{RSM}=\beta^* f^*_\beta \tag{4.249}$$

where β^* and f^*_β are defined in the same way as for the standard $k-\omega$, using Equation 4.85 (p. 63) and Equation 4.91 (p. 64), respectively. The only difference here is that the equation for f^*_β uses a value of 640 instead of 680, as in Equation 4.85 (p. 63).

The constants are

$$\hat{\alpha}_0=\frac{8+C_2}{11},\hat{\beta}_0=\frac{8C_2-2}{11},\hat{\gamma}_0=\frac{60C_2-4}{55}$$

$$C_1=1.8,C_2=0.52$$

The above formulation does not require viscous damping functions to resolve the near-wall sublayer. However, inclusion of the viscous damping function [705] (p. 1094) could improve model predictions for certain flows. This results in the following changes:

$$\hat{\alpha}=\frac{1+\hat{\alpha}_0 Re_T/R_k}{1+Re_T/R_k}$$

$$\hat{\beta}=\hat{\beta}_0\frac{Re_T/R_k}{1+Re_T/R_k}$$

$$\hat{\gamma}=\hat{\gamma}_0\frac{0.007+Re_T/R_k}{1+Re_T/R_k}$$

$$C_1=1.8\left[\frac{5/3+Re_T/R_k}{1+Re_T/R_k}\right]$$

where $\hat{\alpha}$, $\hat{\beta}$, and $\hat{\gamma}$ would replace $\hat{\alpha}_0$, $\hat{\beta}_0$, and $\hat{\gamma}_0$ in Equation 4.247 (p. 102). The constants are

$$R_\beta=12,R_k=12,R_\omega=6.20$$

Inclusion of the low-Re viscous damping is controlled by enabling **Low-Re Corrections** under **k-omega Options** in the Viscous Model Dialog Box.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

4.11.4.5. Stress-BSL Model

The stress-BSL model solves the scale equation from the baseline (BSL) k-ω model, and thus removes the free-stream sensitivity observed with the stress-omega model. It uses the same relation (Equation 4.247 (p. 102)) for the pressure-strain correlation in the Reynolds stress equations as the stress-omega model, but without low-Reynolds number corrections. The coefficient β^*_{RSM} is equal to β^*.

There is also an option **GEKO** which allows the combination of the EARSM with the Generalized $k-\omega$ (GEKO) Model (see Generalized k-ω (GEKO) Model (p. 70) for more information on the GEKO model).

4.11.5. Effects of Buoyancy on Turbulence

The production terms due to buoyancy are modeled as

$$G_{ij} = -\rho\beta\left(g_i\overline{u_j'\theta} + g_j\overline{u_i'\theta}\right) \tag{4.250}$$

$$\overline{U_i\theta} = \frac{\mu_t}{Pr_t}\left(\frac{\partial T}{\partial x_i}\right) \tag{4.251}$$

where Pr_t is the turbulent Prandtl number for energy, with a default value of 0.85.

Using the definition of the coefficient of thermal expansion, β, given by Equation 4.65 (p. 58), the following expression is obtained for G_{ij} for ideal gases:

$$G_{ij} = -\frac{\mu_t}{\rho Pr_t}\left(g_i\frac{\partial \rho}{\partial x_j} + g_j\frac{\partial \rho}{\partial x_i}\right) \tag{4.252}$$

4.11.6. Modeling the Turbulence Kinetic Energy

In general, when the turbulence kinetic energy is needed for modeling a specific term, it is obtained by taking the trace of the Reynolds stress tensor:

$$k = \frac{1}{2}\overline{u_i'u_i'} \tag{4.253}$$

As described in Wall Boundary Conditions (p. 105), an option for the Reynolds stress models based on the ε-equation is available in Ansys Fluent to solve a transport equation for the turbulence kinetic energy in order to obtain boundary conditions for the Reynolds stresses. In this case, the following model equation is used:

$$\begin{aligned}\frac{\partial}{\partial t}\left(\rho k\right) + \frac{\partial}{\partial x_i}\left(\rho k u_i\right) &= \frac{\partial}{\partial x_j}\left[\left(\mu + \frac{\mu_t}{\sigma_k}\right)\frac{\partial k}{\partial x_j}\right] + \frac{1}{2}\left(P_{ii} + G_{ii}\right) \\ &\quad - \rho\varepsilon\left(1 + 2M_t^2\right) + S_k\end{aligned} \tag{4.254}$$

where $\sigma_k = 1.0$ and S_k is a user-defined source term. Equation 4.254 (p. 103) is obtainable by contracting the modeled equation for the Reynolds stresses (Equation 4.226 (p. 98)). As one might expect, it is essentially identical to Equation 4.39 (p. 50) used in the standard k-ε model.

Although Equation 4.254 (p. 103) is solved, the values of k obtained are used only for boundary conditions. In every other case, k is obtained from Equation 4.253 (p. 103). This is a minor point, however, since the values of k obtained with either method should be very similar.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

4.11.7. Modeling the Dissipation Rate

For Reynolds stress models based on the ε-equation, the dissipation tensor ε_{ij} is modeled as

$$\varepsilon_{ij}=\frac{2}{3}\delta_{ij}\left(\rho\varepsilon+Y_M\right) \tag{4.255}$$

where $Y_M=2\rho\varepsilon M_t^2$ is an additional "dilatation dissipation" term according to the model by Sarkar [563] (p. 1085). The turbulent Mach number in this term is defined as

$$M_t=\sqrt{\frac{k}{a^2}} \tag{4.256}$$

where a ($\equiv\sqrt{\gamma RT}$) is the speed of sound. This compressibility modification is available when the compressible form of the ideal gas law is used.

The scalar dissipation rate, ε, is computed with a model transport equation similar to that used in the standard k-ε model:

$$\frac{\partial}{\partial t}\left(\rho\varepsilon\right)+\frac{\partial}{\partial x_i}\left(\rho\varepsilon u_i\right)=\frac{\partial}{\partial x_j}\left[\left(\mu+\frac{\mu_t}{\sigma_\varepsilon}\right)\frac{\partial\varepsilon}{\partial x_j}\right]+C_{\varepsilon1}\frac{1}{2}\left[P_{ii}+C_{\varepsilon3}G_{ii}\right]\frac{\varepsilon}{k}-C_{\varepsilon2}\rho\frac{\varepsilon^2}{k}+S_\varepsilon \tag{4.257}$$

where $\sigma_\varepsilon=1.0$, $C_{\varepsilon1}=1.44$, $C_{\varepsilon2}=1.92$, $C_{\varepsilon3}$ is evaluated as a function of the local flow direction relative to the gravitational vector, as described in Effects of Buoyancy on Turbulence in the k-ε Models (p. 58), and S_ε is a user-defined source term.

In the case when the Reynolds stress model is coupled with the ω- or BSL-equation, the dissipation tensor ε_{ij} is modeled as

$$\varepsilon_{ij}=\frac{2}{3}\delta_{ij}\rho\beta^*_{RSM}k\omega \tag{4.258}$$

where β^*_{RSM} is defined in the corresponding section of Modeling the Pressure-Strain Term (p. 99). For the stress-omega model, the specific dissipation rate ω is computed in the same way as for the standard k-ω model using Equation 4.72 (p. 62), whereas Equation 4.102 (p. 65) from the baseline (BSL) k-ω model is used for the stress-BSL model.

4.11.8. Modeling the Turbulent Viscosity

For Reynolds stress models based on the ε-equation, the turbulent viscosity, μ_t, is computed similarly to the k-ε models:

$$\mu_t=\rho C_\mu\frac{k^2}{\varepsilon} \tag{4.259}$$

where $C_\mu=0.09$.

The Reynolds stress models based on the ω-equation (stress-omega) and BSL-equation (stress-BSL) use Equation 4.74 (p. 62).

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

4.11.9. Wall Boundary Conditions

The RSM model in Ansys Fluent requires boundary conditions for individual Reynolds stresses, $\overline{u_i'u_j'}$, and for the turbulence dissipation rate, ε (or ω if the stress-omega model is used). These quantities can be entered directly or derived from the turbulence intensity and characteristic length (Reynolds Stress Model in the User's Guide).

At walls, Ansys Fluent computes the near-wall values of the Reynolds stresses and ε from wall functions (see Standard Wall Functions (p. 138), Non-Equilibrium Wall Functions (p. 143), and Enhanced Wall Treatment for Momentum and Energy Equations (p. 147)). Ansys Fluent applies explicit wall boundary conditions for the Reynolds stresses by using the log-law and the assumption of equilibrium, disregarding convection and diffusion in the transport equations for the stresses (Equation 4.226 (p. 98)). Using a local coordinate system, where τ is the tangential coordinate, η is the normal coordinate, and λ is the binormal coordinate, the Reynolds stresses at the wall-adjacent cells (assuming standard wall functions or non-equilibrium wall functions) are computed from

$$\frac{\overline{u_\tau'^2}}{k}=1.098,\frac{\overline{u_\eta'^2}}{k}=0.247,\frac{\overline{u_\lambda'^2}}{k}=0.655,-\frac{\overline{u_\tau'u_\eta'}}{k}=0.255 \tag{4.260}$$

To obtain k, Ansys Fluent solves the transport equation of Equation 4.254 (p. 103). For reasons of computational convenience, the equation is solved globally, even though the values of k therefore computed are needed only near the wall; in the far field k is obtained directly from the normal Reynolds stresses using Equation 4.253 (p. 103). By default, the values of the Reynolds stresses near the wall are fixed using the values computed from Equation 4.260 (p. 105), and the transport equations in Equation 4.226 (p. 98) are solved only in the bulk flow region.

Alternatively, the Reynolds stresses can be explicitly specified in terms of wall-shear stress, instead of k:

$$\frac{\overline{u_\tau'^2}}{u_\tau^2}=5.1,\frac{\overline{u_\eta'^2}}{u_\tau^2}=1.0,\frac{\overline{u_\lambda'^2}}{u_\tau^2}=2.3,-\frac{\overline{u_\tau'u_\eta'}}{u_\tau^2}=1.0 \tag{4.261}$$

where u_τ is the friction velocity defined by $u_\tau \equiv \sqrt{\frac{\tau_w}{\rho}}$, where τ_w is the wall-shear stress. When this option is chosen, the k transport equation is not solved.

When using enhanced wall treatments as the near-wall treatment, Ansys Fluent applies zero flux wall boundary conditions to the Reynolds stress equations.

4.11.10. Convective Heat and Mass Transfer Modeling

With the Reynolds stress model in Ansys Fluent, turbulent heat transport is modeled using the concept of Reynolds' analogy to turbulent momentum transfer. The "modeled" energy equation is therefore given by the following:

$$\frac{\partial}{\partial t}\left(\rho E\right)+\frac{\partial}{\partial x_i}\left[u_i\left(\rho E+p\right)\right]=\frac{\partial}{\partial x_j}\left[\left(k+\frac{c_p\mu_t}{Pr_t}\right)\frac{\partial T}{\partial x_j}+u_i\left(\tau_{ij}\right)_{eff}\right]+S_h \tag{4.262}$$

where E is the total energy and $\left(\tau_{ij}\right)_{eff}$ is the deviatoric stress tensor, defined as

$$\left(\tau_{ij}\right)_{eff}=\mu_{eff}\left(\frac{\partial u_j}{\partial x_i}+\frac{\partial u_i}{\partial x_j}\right)-\frac{2}{3}\mu_{eff}\frac{\partial u_k}{\partial x_k}\delta_{ij}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

The term involving $\left(\tau_{ij}\right)_{eff}$ represents the viscous heating, and is always computed in the density-based solvers. It is not computed by default in the pressure-based solver, but it can be enabled in the Viscous Model Dialog Box. The default value of the turbulent Prandtl number is 0.85. You can change the value of Pr_t in the Viscous Model Dialog Box.

Turbulent mass transfer is treated similarly, with a default turbulent Schmidt number of 0.7. This default value can be changed in the Viscous Model Dialog Box.

4.12. Scale-Adaptive Simulation (SAS) Model

This section describes the theory behind the Scale-Adaptive Simulation (SAS) model. For details about using the model in Ansys Fluent, see Setting Up Scale-Adaptive Simulation (SAS) Modeling in the User's Guide.

For more information, see the following sections:

4.12.1. Overview

4.12.2. Transport Equations for the SST-SAS Model

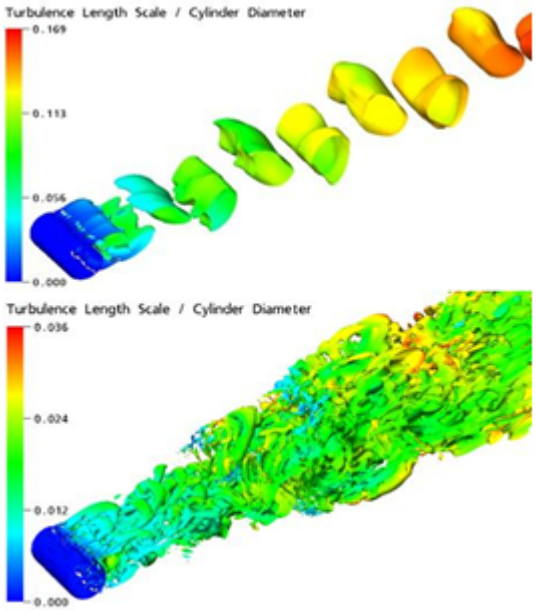
4.12.3. SAS with Other ω-Based Turbulence Models

4.12.1. Overview

The Scale-Adaptive Simulation (SAS) is an improved URANS formulation, which allows the resolution of the turbulent spectrum in unstable flow conditions. Figure 4.6: Resolved Structures for Cylinder in Cross Flow (top: URANS; bottom: SST-SAS) (p. 107) shows isosurfaces where $S-\Omega=10^5 1/s^2$ for a cylinder in cross flow ($Re=3.6*10^6$), calculated using the SST model (URANS) and the SST-SAS model. The URANS simulation produces only the large-scale unsteadiness, whereas the SST-SAS model adjusts to the already resolved scales in a dynamic way and allows the development of a turbulent spectrum in the detached regions.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Figure 4.6: Resolved Structures for Cylinder in Cross Flow (top: URANS; bottom: SST-SAS)



The SAS concept is based on the introduction of the von Kármán length-scale into the turbulence scale equation. The information provided by the von Kármán length-scale allows SAS models to dynamically adjust to resolved structures in a URANS simulation, which results in an LES-like behavior in unsteady regions of the flow field. At the same time, the model provides standard RANS capabilities in stable flow regions.

For information about the synthetic turbulence generation on inlets, refer to Inlet Boundary Conditions for Scale Resolving Simulations (p. 126).

4.12.2. Transport Equations for the SST-SAS Model

From a fundamental standpoint, all the currently used two-equation models suffer from lack of an underlying exact transport equation, which can serve as a guide for the model development on a term by term basis. The reason for this deficiency lies in the fact that the exact equation for ε does not describe the large scales, but the dissipative scales. The goal of a two-equation model is however the modeling of the influence of the large scale motions on the mean flow. Due to the lack of an exact equation, the ε- and the ω-equations are modeled in analogy with the equation for the turbulent kinetic energy, k, using purely heuristic arguments. A more consistent approach for formulating a scale-equation has been developed by Rotta (1968, 1972). Instead of using purely heuristic and dimensional arguments, Rotta formulated an exact transport equation for turbulent kinetic energy times length scale, kL. Rotta's equation represents the large scales of turbulence and can therefore serve as a basis for term-by-term modeling. The transport equations for the SST-SAS model as implemented in Ansys Fluent are based on transforming Rotta's approach to k-ω (SST) and defined as:

$$\frac{\partial \rho k}{\partial t}+\frac{\partial}{\partial x_i}\left(\rho u_i k\right)=G_k-\rho c_\mu k\omega+\frac{\partial}{\partial x_j}\left[\left(\mu+\frac{\mu_t}{\sigma_k}\right)\frac{\partial k}{\partial x_j}\right] \tag{4.263}$$

$$\begin{aligned}\frac{\partial \rho \omega}{\partial t}+\frac{\partial}{\partial x_i}\left(\rho u_i \omega\right)= \;& \alpha\frac{\omega}{k}G_k-\rho\beta\omega^2+Q_{SAS}+\frac{\partial}{\partial x_i}\left[\left(\mu+\frac{\mu_t}{\sigma_\omega}\right)\frac{\partial \omega}{\partial x_j}\right] \\ & +\left(1-F_1\right)\frac{2\rho}{\sigma_{\omega,2}}\frac{1}{\omega}\frac{\partial k}{\partial x_j}\frac{\partial \omega}{\partial x_j}\end{aligned} \tag{4.264}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

For the detailed derivation see Egorov and Menter [428] (p. 1077). The transport equations of the SST-SAS model Equation 4.263 (p. 107) and Equation 4.264 (p. 107) differ from those of the SST RANS model [425] (p. 1077) by the additional SAS source term Q_{SAS} in the transport equation for the turbulence eddy frequency ω, Equation 4.264 (p. 107). In Equation 4.264 (p. 107) $\sigma_{\omega,2}$ is the σ_ω value for the k-ε regime of the SST model.

The additional source term Q_{SAS} reads (see Egorov and Menter [428] (p. 1077) for details):

$$Q_{SAS}=max\left[\rho\eta_2\kappa S^2\left(\frac{L}{L_{v\kappa}}\right)^2-C\cdot\frac{2\rho k}{\sigma_\Phi}max\left(\frac{1}{\omega^2}\frac{\partial\omega}{\partial x_j}\frac{\partial\omega}{\partial x_j},\frac{1}{k^2}\frac{\partial k}{\partial x_j}\frac{\partial k}{\partial x_j}\right),0\right] \tag{4.265}$$

This SAS source term originates from a second order derivative term in Rotta's transport equations [427] (p. 1077). The model parameters in the SAS source term Equation 4.265 (p. 108) are

$$\eta_2=3.51$$
$$\sigma_\Phi=2/3$$
$$C=2$$

Here L is the length scale of the modeled turbulence

$$L=\sqrt{k}/\left(c_\mu^{1/4}\cdot\omega\right) \tag{4.266}$$

and the von Kármán length scale L_{vK} is a three-dimensional generalization of the classic boundary layer definition $L_{BL}^{vK}=\kappa\frac{\partial U}{\partial y}\Big/\frac{\partial^2 U}{\partial y^2}$

$$L_{vK}=\kappa\left|\frac{U'}{U''}\right| \tag{4.267}$$

The first velocity derivative $\frac{\partial U}{\partial y}$ is represented in L_{vK} by U', which is equal to S, a scalar invariant of the strain rate tensor S_{ij}:

$$U'=S=\sqrt{2\cdot S_{ij}S_{ij}};\quad S_{ij}=\frac{1}{2}\left(\frac{\partial U_i}{\partial x_j}+\frac{\partial U_j}{\partial x_i}\right) \tag{4.268}$$

Note, that the same S also directly participates in Q_{SAS} (Equation 4.265 (p. 108)) and in the turbulence production term $P_k=\mu_t S^2$. The second velocity derivative U'' is generalized to 3D using the magnitude of the velocity Laplacian:

$$U''=\sqrt{\frac{\partial^2 U_i}{\partial x_k^2}\frac{\partial^2 U_i}{\partial x_j^2}} \tag{4.269}$$

So defined L and L_{vK} are both equal to (κy) in the logarithmic part of the boundary layer, where $\kappa=0.41$ is the von Kármán constant.

The model also provides a direct control of the high wave number damping. This is realized by a lower constraint on the L_{vK} value in the following way:

$$L_{vK}=max\left(\kappa\left|\frac{U'}{U''}\right|,C_S\sqrt{\frac{\kappa\eta_2}{(\beta/c_\mu)-\alpha}}\cdot\Delta\right),\Delta=\Omega_{CV}^{1/3} \tag{4.270}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

This limiter is proportional to the mesh cell size Δ, which is calculated as the cubic root of the control volume size Ω_{CV}. The purpose of this limiter is to control damping of the finest resolved turbulent fluctuations. The structure of the limiter is derived from analyzing the equilibrium eddy viscosity of the SST-SAS model [428] (p. 1077).

4.12.3. SAS with Other ω-Based Turbulence Models

The SAS approach can be combined not only with the SST model, but also with several other ω-based turbulence models:

- Standard k-ω, BSL k-ω, and Generalized k-ω (GEKO)
- Transition SST
- ω-based Reynolds stress model (RSM) and Explicit Algebraic Reynolds stress model (EARSM)

Adding the SAS capability to one of these models will enable the additional source term Q_{SAS} (as defined in Equation 4.265 (p. 108)) in the ω-based equation. If the SAS capability is combined with the BSL (BSL k-ω, Stress-BSL, WJ-BSL-EARSM), then the coefficients α and β in Equation 4.270 (p. 108) are the model coefficients of the BSL model. If combined with GEKO, then α and β are determined from the GEKO model coefficients. For all other combinations, the model coefficients of the SST model are used. Furthermore, in the SAS term these coefficients do not include the compressibility effects.

4.13. Detached Eddy Simulation (DES)

This section describes the theory behind the Detached Eddy Simulation (DES) model. Information is presented in the following sections:

4.13.1. Overview
4.13.2. DES with the Spalart-Allmaras Model
4.13.3. DES with the Realizable k-ε Model
4.13.4. DES with the BSL or SST k-ω Model
4.13.5. DES with the Transition SST Model
4.13.6. Improved Delayed Detached Eddy Simulation (IDDES)

For details about using the model in Ansys Fluent, see Modeling Turbulence and Setting Up the Detached Eddy Simulation Model in the User's Guide.

4.13.1. Overview

Ansys Fluent offers five different models for the detached eddy simulation: the Spalart-Allmaras model, the realizable k-ε model, the BSL k-ω model, the SST k-ω model, and the Transition SST model.

In the DES approach, the unsteady RANS models are employed in the boundary layer, while the LES treatment is applied to the separated regions. The LES region is normally associated with the core turbulent region where large unsteady turbulence scales play a dominant role. In this region, the DES

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

models recover LES-like subgrid models. In the near-wall region, the respective RANS models are recovered.

DES models have been specifically designed to address high Reynolds number wall bounded flows, where the cost of a near-wall resolving Large Eddy Simulation would be prohibitive. The difference with the LES model is that it relies only on the required RANS resolution in the boundary layers. The application of DES, however, may still require significant CPU resources and therefore, as a general guideline, it is recommended that the conventional turbulence models employing the Reynolds-averaged approach be used for most practical calculations.

The DES models, often referred to as the hybrid LES/RANS models, combine RANS modeling with LES for applications such as high-Re external aerodynamics simulations. In Ansys Fluent, the DES model is based on the one-equation Spalart-Allmaras model, the realizable k-ε model, the BSL k-ω model, the SST k-ω model, and the Transition SST model. The computational costs, when using the DES models, is less than LES computational costs, but greater than RANS.

For information about the synthetic turbulence generation on inlets, refer to Inlet Boundary Conditions for Scale Resolving Simulations (p. 126).

4.13.2. DES with the Spalart-Allmaras Model

The standard Spalart-Allmaras model uses the distance to the closest wall as the definition for the length scale d, which plays a major role in determining the level of production and destruction of turbulent viscosity (Equation 4.20 (p. 46), Equation 4.26 (p. 47), and Equation 4.29 (p. 47)). The DES model, as proposed by Shur et al. [590] (p. 1087) replaces d everywhere with a new length scale $\tilde{d}$, defined as

$$\tilde{d} = \min\left(d, C_{des}\Delta_{\max}\right) \tag{4.271}$$

where $\Delta_{\max}$ is the grid spacing, which in the case of a rectilinear hexahedral cell is the maximum edge length (for other cell types and/or conditions, an extension of this concept is used). The empirical constant C_{des} has a value of 0.65.

For a typical RANS grid with a high aspect ratio in the boundary layer, and where the wall-parallel grid spacing usually exceeds δ, where δ is the size of the boundary layer, Equation 4.271 (p. 110) will ensure that the DES model is in the RANS mode for the entire boundary layer. However, in case of an ambiguous grid definition, where $\Delta_{\max} < \delta$, the DES limiter can activate the LES mode inside the boundary layer, where the grid is not fine enough to sustain resolved turbulence. Therefore, a new formulation [619] (p. 1089) of DES is available in Ansys Fluent to preserve the RANS mode throughout the boundary layer. This is known as the delayed option or DDES for delayed DES.

The DES length scale $\tilde{d}$ is re-defined according to:

$$\tilde{d} = d - f_d \max\left(0, d - C_{des}\Delta_{\max}\right) \tag{4.272}$$

where f_d is given by:

$$f_d = 1 - \tanh\left(\left(8r_d\right)^3\right) \tag{4.273}$$

and

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$r_d=\frac{\tilde{\nu}}{\sqrt{U_{i,j}U_{i,j}}\,\kappa^2 d^2};\quad \kappa=0.41 \tag{4.274}$$

This formulation is the default settings.

4.13.3. DES with the Realizable k-ε Model

This DES model is similar to the Realizable k-ε model discussed in Realizable k-ε Model (p. 54), with the exception of the dissipation term in the k equation. In the DES model, the Realizable k-ε RANS dissipation term is modified such that:

$$Y_k=\frac{\rho k^{\frac{3}{2}}}{l_{des}} \tag{4.275}$$

where

$$l_{des}=min\left(l_{rke},l_{les}\right) \tag{4.276}$$

$$l_{rke}=\frac{k^{\frac{3}{2}}}{\varepsilon} \tag{4.277}$$

$$l_{les}=C_{des}\Delta_{\max} \tag{4.278}$$

where C_{des} is a calibration constant used in the DES model and has a value of 0.61 and $\Delta_{\max}$ is the grid spacing, which in the case of a rectilinear hexahedral cell is the maximum edge length (for other cell types and/or conditions, an extension of this concept is used).

For the case where $l_{des}=l_{rke}$, you will obtain an expression for the dissipation of the k formulation for the Realizable k-ε model (Realizable k-ε Model (p. 54)): $Y_k=\rho\varepsilon$. Similarly to the Spalart-Allmaras model, the delayed concept can be applied as well to the Realizable DES model to preserve the RANS mode throughout the boundary layer. The DES length l_{des} in Equation 4.275 (p. 111) is redefined such that

$$l_{des}=l_{rke}-f_d\max\left(0,l_{rke}-C_{des}\Delta_{\max}\right) \tag{4.279}$$

Note:

In Equation 4.279 (p. 111) f_d is used as defined for the Spalart-Allmaras model in Equation 4.273 (p. 110) with the exception that the value of the constant is changed from 8 to 20 and $\tilde{\nu}$ is replaced with $(\nu_t+\nu)$ in the calculation of r_d in Equation 4.274 (p. 111).

4.13.4. DES with the BSL or SST k-ω Model

The dissipation term of the turbulent kinetic energy (see Modeling the Turbulence Dissipation (p. 63)) is modified for the DES turbulence model as described in Menter's work [429] (p. 1078) such that

$$Y_k=\rho\beta^* k\omega F_{DES} \tag{4.280}$$

where F_{DES} is expressed as

$$F_{DES}=max\left(\frac{L_t}{C_{des}\Delta_{\max}},1\right) \tag{4.281}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

where C_{des} is a calibration constant used in the DES model and has a value of 0.61, and $\Delta_{\max}$ is the grid spacing, which in the case of a rectilinear hexahedral cell is the maximum edge length (for other cell types and/or conditions, an extension of this concept is used).

The turbulent length scale is the parameter that defines this RANS model:

$$L_t = \frac{\sqrt{k}}{\beta^* \omega} \tag{4.282}$$

The DES-BSL / SST model also offers the option to "protect" the boundary layer from the limiter (delayed option). This is achieved with the help of the zonal formulation of the BSL / SST model. F_{DES} is modified according to

$$F_{DES} = max\left(\frac{L_t}{C_{des}\Delta_{\max}} \left(1 - F_{SST}\right), 1 \right) \tag{4.283}$$

with $F_{SST} = 0, F_1, F_2$, where F_1 and F_2 are the blending functions of the BSL / SST model. Alternatively, the DDES shielding function or the IDDES function can be selected [219] (p. 1065), [619] (p. 1089). The default setting is to use DDES.

Note:

In the DDES and IDDES implementations, f_d is used as defined for the Spalart-Allmaras model in Equation 4.273 (p. 110) with the exception that the value of the constant is changed from 8 to 20.

The default setting is to use DDES. The F_{DDES} blending function is given by:

$$F_{DDES} = \tanh\left[\left(C_{d1} r_d \right)^{C_{d2}} \right] \tag{4.284}$$

where C_{d1}=20, C_{d2}=3, and

$$r_d = \frac{\nu_t + \nu}{\kappa^2 y^2 \sqrt{0.5 \cdot \left(S^2 + \Omega^2 \right)}} \tag{4.285}$$

Here S is the magnitude of the strain rate tensor, Ω is the magnitude of vorticity tensor, y is the wall distance, and κ=0.41.

4.13.5. DES with the Transition SST Model

The Transition SST model can be combined with the DES approach. The dissipation term of the transport equation for the turbulence kinetic energy will be modified in the same way as for the SST k-ω turbulence model (see Equation 4.280 (p. 111)). All shielding functions outlined in DES with the BSL or SST k-ω Model (p. 111) are also available for DES with the Transition SST model.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

4.13.6. Improved Delayed Detached Eddy Simulation (IDDES)

4.13.6.1. Overview of IDDES

The Improved Delayed Detached Eddy Simulation (IDDES) model ([589] (p. 1087), [219] (p. 1065)) is a hybrid RANS-LES model (consisting of a combination of various new and existing techniques) that provides a more flexible and convenient scale-resolving simulation (SRS) model for high Reynolds number flows. Since the model formulation is relatively complex and the application of the model is non-trivial, it is recommended that you consult the original publications of Shur et al. [589] (p. 1087) and Gritskevich et al. [219] (p. 1065).

The IDDES model has the following goals in addition to the formulation of the standard DES model:

- Provide shielding against Grid Induced Separation (GIS), similar to the DDES model ([619] (p. 1089)).
- Allow the model to run in Wall-Modeled LES (WMLES) mode in case unsteady inlet conditions are provided to simulate wall boundary layers in unsteady mode. The IDDES model is designed to allow the LES simulation of wall boundary layers at much higher Reynolds numbers than standard LES models. As an alternative to unsteady inlet conditions, unsteadiness could also be triggered by an obstacle (such as a backward facing step, or a rib inside or upstream of the boundary layer).

The IDDES model implemented in Ansys Fluent is based on the BSL / SST model ([425] (p. 1077)) with the application of IDDES modifications as given in [219] (p. 1065). Similar to DES, the k-equation of the BSL / SST model is modified to include information on the local grid spacing. In case the grid resolution is sufficiently fine, the model will switch to LES mode. However, the goal is to cover stable boundary layers (meaning no unsteady inlet conditions and no upstream obstacles generating unsteadiness) in RANS mode. In order to avoid affecting the BSL / SST model under such conditions, the IDDES function provides shielding similar to the DDES model, meaning it attempts to keep the boundary layer in steady RANS mode even under grid refinement.

If you want to resolve the wall boundary layer in WMLES mode, unsteady inlet conditions need to be provided (see Vortex Method (p. 127) or Spectral Synthesizer (p. 129)). The model can also be run with the Embedded LES (ELES) option (see Embedded Large Eddy Simulation (ELES) (p. 132)) and the IDDES option (see Detached Eddy Simulation (DES) in the User's Guide) in Ansys Fluent.

4.13.6.2. IDDES Model Formulation

The IDDES-BSL / SST model is based on modifying the sink term in the k-equation of the BSL / SST model. (The ω-equation remains unmodified.)

$$\begin{aligned}\frac{\partial \rho k}{\partial t}+\nabla\cdot(\overline{\rho}\tilde{U}k)&=\nabla\cdot[(\mu+\frac{\mu_T}{\sigma_{k3}})\nabla k]+P_k-\rho\beta^*k\omega F_{IDDES}\\ \frac{\partial \rho \omega}{\partial t}+\nabla\cdot(\overline{\rho}\tilde{U}\omega)&=\nabla\cdot[(\mu+\frac{\mu_T}{\sigma_{\omega 3}})\nabla \omega]+(1-F_1)2\rho\frac{\nabla k\cdot\nabla\omega}{\sigma_{\omega 2}\omega}+\alpha_3\frac{\omega}{k}P_k-\beta_3\rho\omega^2\end{aligned} \tag{4.286}$$

$$F_{IDDES}=\frac{l_{RANS}}{l_{IDDES}} \tag{4.287}$$

where $l_{RANS}=\sqrt{k}\,/\,(\beta^*\omega)$.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

In Equation 4.287 (p. 113), F_{IDDES} is based on the RANS turbulent length scale and the LES grid length scales. Note that the sub-grid length scale Δ differs from the DES formulation. In the IDDES formulation, a more general formulation for Δ is used that combines local grid scales and the wall distance d_w.

The complete formulation is relatively complex and has been implemented as published by [219] (p. 1065), except for the value of $h_{\max}$; while this variable is defined as the maximum edge length in the case of a rectilinear hexahedral cell in Ansys Fluent, for other cell types and/or conditions an extension of this concept is used.

4.14. Shielded Detached Eddy Simulation (SDES)

Shielded Detached Eddy Simulation (SDES) is a hybrid RANS-LES turbulence model family. Starting from the DDES-SST model (see DES with the BSL or SST k-ω Model (p. 111) for details), it provides substantial improvements:

- a shielding function that protects the RANS wall boundary layer region against influences from the LES model
- improved definition of the grid length scale, for faster "transition" from RANS to LES in separating shear layers

The improved shielding function offers nearly perfect boundary layer shielding against a premature switch to the LES model. Such premature switching could otherwise cause strong deterioration of the RANS capabilities of the model in attached boundary layers [619] (p. 1089).

Note:

No special calibration has been performed for the combination of SDES with Transition SST and the Intermittency Transition model.

Note that the activation of the LES term in the freestream in any hybrid RANS-LES model can affect the decay of the freestream turbulence, which in turn affects the transition location.

For an example demonstrating the effects of the SDES model, see SDES and SBES Example (p. 118).

4.14.1. Shielding Function

The starting point for the development of the Shielded Detached Eddy Simulation (SDES) shielding function is the existing Delayed Detached Eddy Simulation (DDES) models (see Detached Eddy Simulation (DES) (p. 109)). In DDES, the primary goal is to model the attached and mildly separated boundary layers in RANS mode and then switch to LES mode for separated (detached) shear layers. Starting from a given RANS model, a DDES formulation is achieved in two-equation models by a modification of the sink term in the k-equation. In the k-equation of the SST model, the modification is reformulated as follows:

$$\varepsilon_{\mathrm{DDES}} = -\beta^* \rho k \omega F_{\mathrm{DDES}} \tag{4.288}$$

where

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$F_{\mathrm{DDES}}=\left[\max\left(\frac{L_t}{C_{\mathrm{DES}}\Delta_{\max}}\left(1-f_{\mathrm{DDES}}\right),1\right)-1\right] \tag{4.289}$$

In the previous equation, L_t is the turbulence length scale, $\Delta_{\max}$ is the maximum edge length of the local cell, f_{DDES} is the DDES shielding function and C_{DES} is a coefficient. When F_{DDES} equals zero, the RANS model is recovered. This happens when one or both of the following conditions are satisfied:

$$f_{\mathrm{DDES}}=1 \tag{4.290}$$

and/or

$$\frac{L_t}{C_{\mathrm{DES}}\Delta_{\max}}<1 \tag{4.291}$$

The function f_{DDES} is designed to protect attached boundary layers from the switch to the LES regime. The limit at which the boundary layer becomes affected depends on the ratio of the grid spacing to the boundary layer thickness (for example, $r=\Delta_{\max}/\delta$). Without shielding (that is, when $f_{\mathrm{DDES}}=0$) the limit is approximately $r=1$, so that at $r<1$ the solution in the boundary layer becomes affected by the DES sink term. With the shielding function as given in Detached Eddy Simulation (DES) (p. 109), the limit for r is reduced to 0.2-0.3, depending on the pressure gradient. Violating these limits has severe consequences, as the RANS solution will become compromised, and you might not be aware of it. DES / DDES methods were originally developed for aircraft simulations, where the boundary layer is typically thin relative to the dimensions of the aircraft / wing. Under those conditions (and with diligent mesh generation practices), the previously described limits for r can typically be ensured. However, it is harder to ensure these limits for general industrial applications, due to lower Reynolds numbers, less control over the grid spacing, and more complex interactions between components.

The first goal of the development of the SDES method was therefore the reformulation of the function f_{DDES} to achieve more reliable shielding, meaning the ability to maintain RANS boundary layers even under much more severe mesh refinement in attached boundary layers than is possible for DDES. For this purpose, a new unpublished and Ansys proprietary shielding function called f_{SDES} was developed, which provides substantially safer shielding than the f_{DDES} functions currently used. The resulting sink term is equivalent to the one from the DDES model:

$$\varepsilon_{\mathrm{SDES}}=-\beta^{*}\rho k\omega F_{\mathrm{SDES}} \tag{4.292}$$

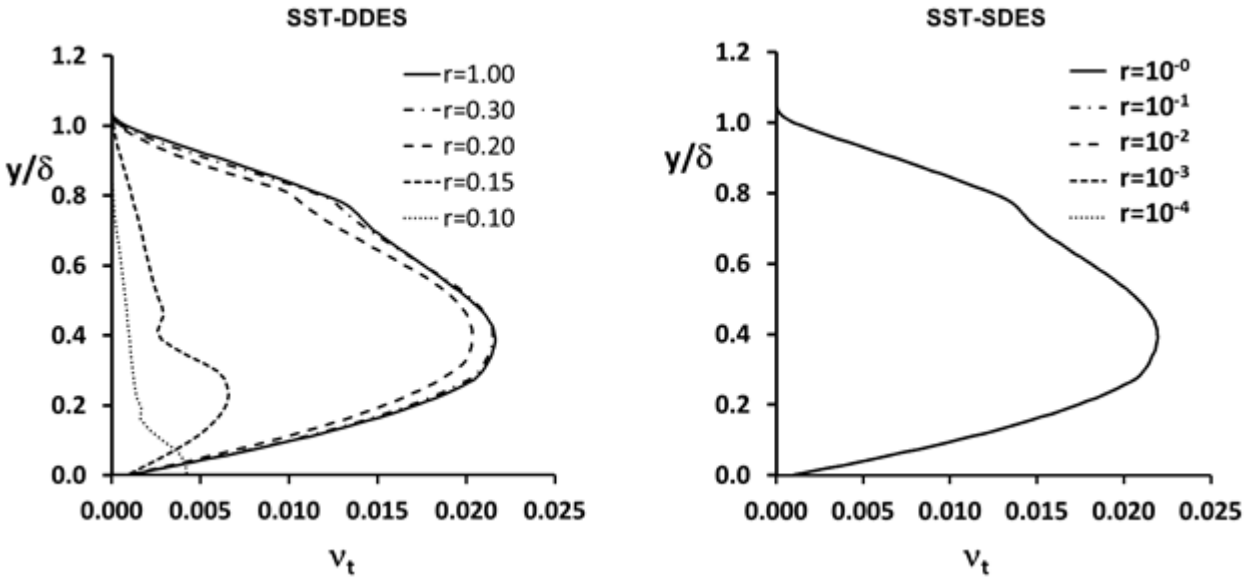
where

$$F_{\mathrm{SDES}}=\left[\max\left(\frac{L_t}{C_{\mathrm{SDES}}\Delta_{\mathrm{SDES}}}\left(1-f_{\mathrm{SDES}}\right),1\right)-1\right] \tag{4.293}$$

Figure 4.7: Eddy Viscosity Profiles (p. 116) shows the eddy viscosities for self-similar zero-pressure-gradient boundary layer simulations under mesh refinement, where $r=\Delta_{\max}/\delta$ (note the different range of the refinement ratio r for the two models). The SDES shielding (on the right) is clearly superior to the DDES shielding, and maintains the RANS boundary layer even under severe mesh refinement.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Figure 4.7: Eddy Viscosity Profiles



4.14.2. LES Mode of SDES

After the SDES model switches to LES mode, the additional source term is activated and reduces the eddy viscosity to a level comparable to a conventional LES model. The level of the achieved eddy viscosity can be estimated by imposing equilibrium on the source terms of the underlying RANS model (in this case, the SST model). This means that the convection and diffusion terms in both the k- and ω-equations are neglected, and the source and sink terms are equated. The result is:

$$\nu_t = = \left(\left(\frac{\beta}{\alpha} \right)^{3/4} C_{\mathrm{LES}} \Delta \right)^2 S \tag{4.294}$$

where S is the strain rate, and α and β are the constants of the source and sink terms in the ω-equation, respectively. C_{LES} represents the constants C_{DES} and C_{SDES} used in the different models, and Δ represents the definition of the grid spacing used in the different models. The formulation is obviously equivalent to the classical Smagorinsky model (see Subgrid-Scale Models (p. 120)). The combination of $(\beta/\alpha)^{3/4}\ C_{\mathrm{LES}}$ is equivalent to the constant C_s in the Smagorinsky model. Using the outer constants from the SST model that are relevant for the LES region (α=0.44, β=0.083, C_{DES}=0.61) produces an equivalent C_s=0.175 for the DES / DDES models. This is close to the value of C_s=0.18, which is recommended for the Smagorinsky model for Decaying Isotropic Turbulence (DIT) simulations. This is not surprising, as the LES portion of DES / DDES was calibrated for DIT. However, it is known that the Smagorinsky model requires different calibration constants for DIT and shear turbulence. The value for shear flows is closer to C_s=0.11. Since shear flows are much more relevant for engineering simulations than DIT, it was decided to use a value of C_{SDES}=0.4 in the SDES model, which results in the desired value for the equivalent C_s constant.

The definition of the LES length scale is $\Delta=\Delta_{\max}$ in the DES / DDES models. This definition is problematic, as it can result in overly high levels of eddy viscosity in separating shear layers, where the mesh aspect ratio is typically high (for example, separating flow from a backward facing step, where the spanwise grid spacing is much larger than the grid spacing in the other two directions). This can lead

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

to a slow 'transition' from RANS to LES mode. In order to avoid this issue, the SDES model uses the following formulation for the LES length scale:

$$\Delta_{\mathrm{SDES}}=\max\left(\sqrt[3]{V},0.2\Delta_{\max}\right) \tag{4.295}$$

The first part is the classical length scale based on the volume V of the cell, and the second part ensures a viable limit for very high aspect ratios. This formulation is smaller by a factor of five for high aspect ratio meshes compared to the DES definition.

It is important to note that both the constant C_{DES} / C_{SDES} and the mesh definitions enter the equivalent Smagorinsky model quadratically. For highly stretched meshes, as is typical for separating shear layers, the combined effect of a lower C_{SDES} constant and a smaller grid spacing definition results in a reduction by a factor 60 in the eddy viscosity level for SDES compared to DES / DDES.

It should also be noted that both modifications could equally be applied to the DES / DDES model formulations. However, such changes would severely reduce the shielding properties of the DES / DDES formulation, which are based on the combination of $C_{\mathrm{DES}}\,\Delta_{\max}$. Lowering these values alone or in combination would impair the shielding properties of these models.

4.15. Stress-Blended Eddy Simulation (SBES)

Stress-Blended Eddy Simulation (SBES) is a hybrid RANS-LES turbulence model that uses the same shielding function as the SDES formulation (see Shielded Detached Eddy Simulation (SDES) (p. 114)), and adds the ability to switch (blend) from the underlying RANS model directly to any existing algebraic LES model (again, based on the shielding function). This functionality provides the name for the SBES model.

Note:

No special calibration has been performed for the combination of SBES with transition models (Transition SST, the Intermittency Transition model, or the Algebraic Transition model).

4.15.1. Stress Blending

The shielding function developed within the SDES framework can be extended to achieve a blending on the stress level between RANS and LES formulations. In general terms, this affects the turbulence stress tensor in the following way:

$$\tau_{ij}^{\mathrm{SBES}}=f_{\mathrm{SDES}}\tau_{ij}^{\mathrm{RANS}}+\left(1-f_{\mathrm{SDES}}\right)\tau_{ij}^{\mathrm{LES}} \tag{4.296}$$

where $\tau_{ij}^{\mathrm{RANS}}$ is the RANS portion and τ_{ij}^{LES} is the LES portion of the modeled stress tensor. In cases where both models are based on eddy viscosity concepts, the formulation simplifies to:

$$\nu_{t}^{\mathrm{SBES}}=f_{\mathrm{SDES}}\nu_{t}^{\mathrm{RANS}}+\left(1-f_{\mathrm{SDES}}\right)\nu_{t}^{\mathrm{LES}} \tag{4.297}$$

Such a formulation is only feasible due to the strength of the shielding properties of the shielding function f_{SDES} (as described in the previous section).

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

The main advantage of the SBES model is that it allows you to use a given LES model in the LES portion of the flow, instead of relying on the LES capability of DES-like models. You can also use it to visualize the regions where RANS and LES models are used, in order to check the consistency of your setup. Finally, the SBES model allows you to have a rapid "transition" from RANS to LES in separating shear layers.

In summary, the SBES model allows you to:

- generically combine RANS and LES model formulations
- clearly distinguish between the RANS and the LES regions based on the f_{SDES} function
- utilize a rapid "transition" from RANS to LES in separating shear layers, due to the low stress levels enforced by the LES model

4.15.2. SDES and SBES Example

The following example illustrates the application of the SDES and SBES models to a jet emanating from a nozzle. Figure 4.8: The Computational Domain and Mesh for the Subsonic Jet Flow (p. 118) shows the geometry and mesh. The round jet flow has been experimentally investigated by Viswanathan [672] (p. 1092) at a Mach number of M=0.9 and at a Reynolds number of Re_D=1.3 x 10^6, based on the jet diameter D and on the bulk velocity at the nozzle exit U_0. The computational mesh consisted of 4.1 x 10^6 hexahedral cells.

Figure 4.8: The Computational Domain and Mesh for the Subsonic Jet Flow

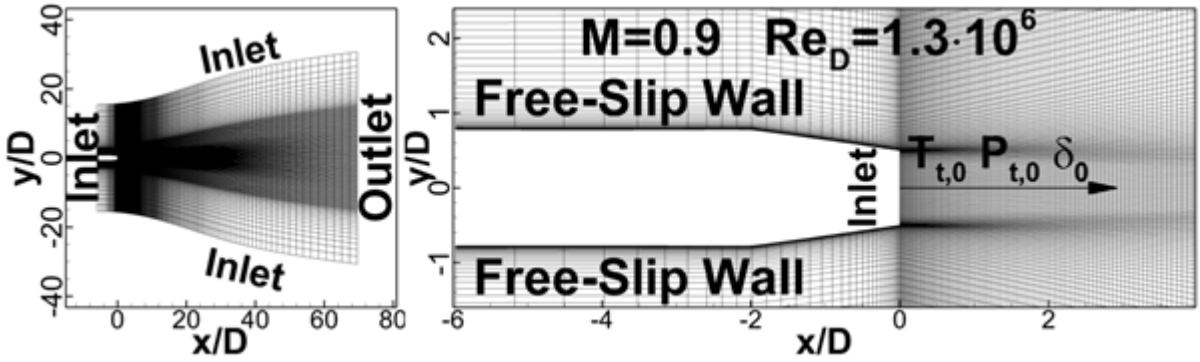


Figure 4.9: Iso-Surfaces of the Q-Criterion Colored with the Velocity Magnitude (p. 119) shows the turbulence structures developing with the different models. The main goal of hybrid RANS-LES models is to "transition" as quickly as possible from the steady RANS mode into the scale resolving three-dimensional LES mode.

As seen from Figure 4.9: Iso-Surfaces of the Q-Criterion Colored with the Velocity Magnitude (p. 119), SDES and SBES provide a relatively rapid transition from the two-dimensional RANS to three-dimensional LES structures. The fastest transition is achieved by the SBES formulation, followed by SDES. DDES yields almost axisymmetric structures, which prevail for a noticeably longer distance from the nozzle.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Figure 4.9: Iso-Surfaces of the Q-Criterion Colored with the Velocity Magnitude

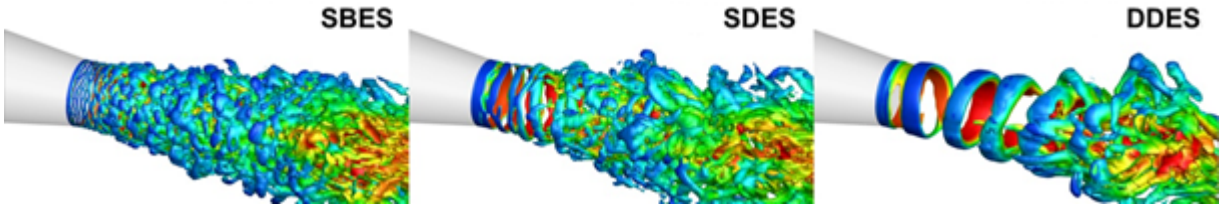
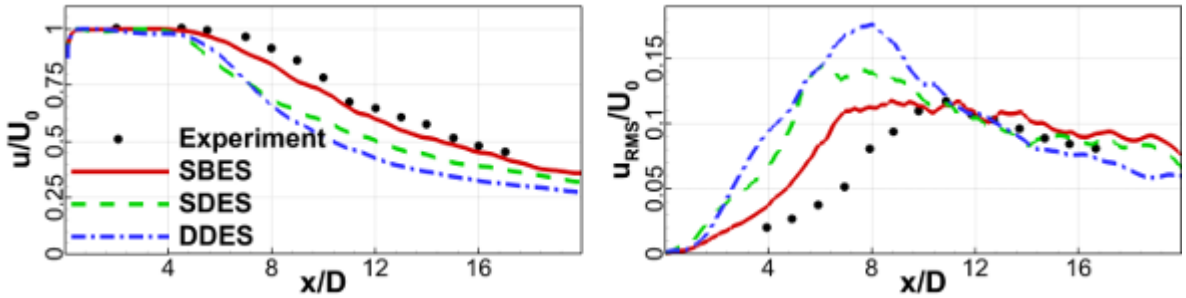


Figure 4.10: Distribution of the Mean (Left) and RMS (Right) Velocity along the Jet Centerline (p. 119) shows the comparison with experimental data. The left plot shows the mean axial velocity along the centerline of the jet, and the right plot shows the corresponding streamwise RMS fluctuations. It is clear from both comparisons that the agreement with the data improves with the models' ability to quickly switch into LES mode.

Figure 4.10: Distribution of the Mean (Left) and RMS (Right) Velocity along the Jet Centerline



4.16. Large Eddy Simulation (LES) Model

This section describes the theory behind the Large Eddy Simulation (LES) model. Information is presented in the following sections:

4.16.1. Overview

4.16.2. Subgrid-Scale Models

4.16.3. Inlet Boundary Conditions for Scale Resolving Simulations

For details about using the model in Ansys Fluent, see Modeling Turbulence and Setting Up the Large Eddy Simulation Model in the User's Guide.

4.16.1. Overview

Turbulent flows are characterized by eddies with a wide range of length and time scales. The largest eddies are typically comparable in size to the characteristic length of the mean flow (for example, shear layer thickness). The smallest scales are responsible for the dissipation of turbulence kinetic energy.

It is possible, in theory, to directly resolve the whole spectrum of turbulent scales using an approach known as direct numerical simulation (DNS). No modeling is required in DNS. However, DNS is not feasible for practical engineering problems involving high Reynolds number flows. The cost required for DNS to resolve the entire range of scales is proportional to Re_t^3, where Re_t is the turbulent Reynolds number. Clearly, for high Reynolds numbers, the cost becomes prohibitive.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

In LES, large eddies are resolved directly, while small eddies are modeled. Large eddy simulation (LES) therefore falls between DNS and RANS in terms of the fraction of the resolved scales. The rationale behind LES can be summarized as follows:

- Momentum, mass, energy, and other passive scalars are transported mostly by large eddies.
- Large eddies are more problem-dependent. They are dictated by the geometries and boundary conditions of the flow involved.
- Small eddies are less dependent on the geometry, tend to be more isotropic, and are consequently more universal.
- The chance of finding a universal turbulence model is much higher for small eddies.

Resolving only the large eddies allows one to use much coarser mesh and larger time-step sizes in LES than in DNS. However, LES still requires substantially finer meshes than those typically used for RANS calculations. In addition, LES has to be run for a sufficiently long flow-time to obtain stable statistics of the flow being modeled. As a result, the computational cost involved with LES is normally orders of magnitudes higher than that for steady RANS calculations in terms of memory (RAM) and CPU time. Therefore, high-performance computing (for example, parallel computing) is a necessity for LES, especially for industrial applications. The main shortcoming of LES lies in the high resolution requirements for wall boundary layers. Near the wall, even the 'large' eddies become relatively small and require a Reynolds number dependent resolution. This limits LES for wall bounded flows to very low Reynolds numbers ($\mathrm{Re}\sim 10^4-10^5$) and limited computational domains.

The following sections give details of the governing equations for LES, the subgrid-scale turbulence models, and the boundary conditions.

4.16.2. Subgrid-Scale Models

The subgrid-scale (SGS) stresses resulting from the filtering operation are unknown, and require modeling. The subgrid-scale turbulence models in Ansys Fluent employ the Boussinesq hypothesis [246] (p. 1067) as in the RANS models, computing subgrid-scale turbulent stresses from

$$\tau_{ij}-\frac{1}{3}\tau_{kk}\delta_{ij}=-2\mu_t\overline{S}_{ij} \tag{4.298}$$

where μ_t is the subgrid-scale turbulent viscosity. The isotropic part of the subgrid-scale stresses τ_{kk} is not modeled, but added to the filtered static pressure term. $\overline{S}_{ij}$ is the rate-of-strain tensor for the resolved scale defined by

$$\overline{S}_{ij}\equiv\frac{1}{2}\left(\frac{\partial\overline{u}_i}{\partial x_j}+\frac{\partial\overline{u}_j}{\partial x_i}\right) \tag{4.299}$$

For compressible flows, it is convenient to introduce the density-weighted (or Favre) filtering operator:

$$\tilde{\varphi}=\frac{\overline{\rho\varphi}}{\overline{\rho}} \tag{4.300}$$

The Favre Filtered Navier-Stokes equation takes the same form as Equation 4.9 (p. 43). The compressible form of the subgrid stress tensor is defined as:

$$\tau_{ij}=\overline{\rho}u_i\tilde{u}_j-\overline{\rho}\tilde{u}_i\tilde{u}_j \tag{4.301}$$

This term is split into its isotropic and deviatoric parts

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$\tau_{ij}=\underbrace{\tau_{ij}-\frac{1}{3}\tau_{kk}\delta_{ij}}_{\text{deviatoric}}+\underbrace{\frac{1}{3}\tau_{kk}\delta_{ij}}_{\text{isotropic}} \tag{4.302}$$

The deviatoric part of the subgrid-scale stress tensor is modeled using the compressible form of the Smagorinsky model:

$$\tau_{ij}-\frac{1}{3}\tau_{kk}\delta_{ij}=-2\mu_t\left(S_{ij}-\frac{1}{3}S_{kk}\delta_{ij}\right) \tag{4.303}$$

As for incompressible flows, the term involving τ_{kk} can be added to the filtered pressure or simply neglected [165] (p. 1062). Indeed, this term can be re-written as $\tau_{kk}=\gamma M^2_{sgs}\overline{p}$ where M_{sgs} is the subgrid Mach number. This subgrid Mach number can be expected to be small when the turbulent Mach number of the flow is small.

Ansys Fluent offers four models for μ_t: the Smagorinsky-Lilly model, the dynamic Smagorinsky-Lilly model, the WALE model, and the dynamic kinetic energy subgrid-scale model.

The subgrid-scale turbulent flux of a scalar, φ, is modeled using a subgrid-scale turbulent Prandtl number by

$$q_j=-\frac{\mu_t}{\sigma_t}\frac{\partial\varphi}{\partial x_j} \tag{4.304}$$

where q_j is the subgrid-scale flux.

The compressible subgrid enthalpy flux term is modeled in the same manner:

$$\overline{\rho}\left(u_i\tilde{h}_s-\tilde{u}_i\tilde{h}_s\right)=-\frac{\mu_{\text{SGS}}C_p}{\text{Pr}_{\text{SGS}}}\frac{\partial\tilde{T}}{\partial x_j} \tag{4.305}$$

where h_s is the sensible enthalpy, μ_{SGS} is a subgrid viscosity, and Pr_{SGS} is a subgrid Prandtl number equal to 0.85.

In the dynamic models, the subgrid-scale turbulent Prandtl number or Schmidt number is obtained by applying the dynamic procedure originally proposed by Germano [199] (p. 1064) to the subgrid-scale flux.

4.16.2.1. Smagorinsky-Lilly Model

This simple model was first proposed by Smagorinsky [600] (p. 1088). In the Smagorinsky-Lilly model, the eddy-viscosity is modeled by

$$\mu_t=\rho L_s^2\left|\overline{S}\right| \tag{4.306}$$

where L_s is the mixing length for subgrid scales and $\left|\overline{S}\right|\equiv\sqrt{2\overline{S}_{ij}\overline{S}_{ij}}$. In Ansys Fluent, L_s is computed using

$$L_s=min\left(\kappa d,C_s\Delta\right) \tag{4.307}$$

where $\kappa=0.41$ is the von Kármán constant, d is the distance to the closest wall, C_s is the Smagorinsky constant, and Δ is the local grid scale. In Ansys Fluent, Δ is computed according to the volume of the computational cell using

$$\Delta=V^{1/3} \tag{4.308}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Lilly derived a value of 0.23 for C_s for homogeneous isotropic turbulence in the inertial subrange. However, this value was found to cause excessive damping of large-scale fluctuations in the presence of mean shear and in transitional flows as near solid boundary, and has to be reduced in such regions. In short, C_s is not a universal constant, which is the most serious shortcoming of this simple model. Nonetheless, a C_s value of around 0.1 has been found to yield the best results for a wide range of flows, and is the default value in Ansys Fluent.

4.16.2.2. Dynamic Smagorinsky-Lilly Model

Germano et al. [199] (p. 1064) and subsequently Lilly [373] (p. 1074) conceived a procedure in which the Smagorinsky model constant, C_s, is dynamically computed based on the information provided by the resolved scales of motion. The dynamic procedure therefore obviates the need for users to specify the model constant C_s in advance.

The concept of the dynamic procedure is to apply a second filter (called the test filter) to the equations of motion. The new filter width $\hat{\Delta}$ is equal to twice the grid filter width Δ. Both filters produce a resolved flow field. The difference between the two resolved fields is the contribution of the small scales whose size is in between the grid filter and the test filter. The information related to these scales is used to compute the model constant. In Ansys Fluent, the variable density formulation of the model is considered as explained below.

At the test filtered field level, the SGS stress tensor can be expressed as:

$$T_{ij}=\overline{\rho\hat{u}_i u_j}-\left(\overline{\rho\hat{u}_i}\overline{\rho\hat{u}_j}/\ \hat{\overline{\rho}}\right) \tag{4.309}$$

Both T_{ij} and τ_{ij} are modeled in the same way with the Smagorinsky-Lilly model, assuming scale similarity:

$$\tau_{ij}=-2C\overline{\rho}\Delta^2\widetilde{S}\left|\left(\widetilde{S}_{ij}-\frac{1}{3}\widetilde{S}_{kk}\delta_{ij}\right)\right. \tag{4.310}$$

$$T_{ij}=-2C\hat{\overline{\rho}}\hat{\Delta}^2\left|\hat{\widetilde{S}}\right|\left(\hat{\widetilde{S}}_{ij}-\frac{1}{3}\hat{\widetilde{S}}_{kk}\delta_{ij}\right) \tag{4.311}$$

In Equation 4.310 (p. 122) and Equation 4.311 (p. 122), the coefficient C is assumed to be the same and independent of the filtering process (note that per Equation 4.307 (p. 121), $C=C_s^2$). The grid filtered SGS and the test-filtered SGS are related by the Germano identity [199] (p. 1064) such that:

$$L_{ij}=T_{ij}-\hat{\tau}_{ij}=\overline{\rho}\hat{\widetilde{u}_i}\widetilde{u}_j-\frac{1}{\hat{\overline{\rho}}}\left(\overline{\rho\hat{\widetilde{u}}_i}\overline{\rho\hat{\widetilde{u}}_j}\right) \tag{4.312}$$

Where L_{ij} is computable from the resolved large eddy field. Substituting the grid-filter Smagorinsky-Lilly model and Equation 4.311 (p. 122) into Equation 4.312 (p. 122), the following expressions can be derived to solve for C with the contraction obtained from the least square analysis of Lilly (1992).

$$C=\frac{\left(L_{ij}-L_{kk}\delta_{ij}/\ 3\right)M_{ij}}{M_{ij}M_{ij}} \tag{4.313}$$

With

$$M_{ij}=-2\left(\hat{\Delta}^2\hat{\overline{\rho}}\left|\hat{\widetilde{S}}\right|\hat{\widetilde{S}}_{ij}-\Delta^2\overline{\rho}\left|\hat{\widetilde{S}}\right|\widetilde{S}_{ij}\right) \tag{4.314}$$

More details of the model implementation in Ansys Fluent and its validation can be found in [305] (p. 1070).

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

The $C_s=\sqrt{C}$ obtained using the dynamic Smagorinsky-Lilly model varies in time and space over a fairly wide range. To avoid numerical instability, both the numerator and the denominator in Equation 4.313 (p. 122) are locally averaged (or filtered) using the test-filter. In Ansys Fluent, C_s is also clipped at zero and 0.23 by default.

4.16.2.3. Wall-Adapting Local Eddy-Viscosity (WALE) Model

In the WALE model [475] (p. 1080), the eddy viscosity is modeled by

$$\mu_t=\rho L_s^2\frac{\left(S_{ij}^d S_{ij}^d\right)^{3/2}}{\left(\overline{S}_{ij}\overline{S}_{ij}\right)^{5/2}+\left(S_{ij}^d S_{ij}^d\right)^{5/4}} \tag{4.315}$$

where L_s and S_{ij}^d in the WALE model are defined, respectively, as

$$L_s=min\left(\kappa d, C_w V^{1/3}\right) \tag{4.316}$$

$$S_{ij}^d=\frac{1}{2}\left(\overline{g}_{ij}^2+\overline{g}_{ji}^2\right)-\frac{1}{3}\delta_{ij}\overline{g}_{kk}^2, \overline{g}_{ij}=\frac{\partial\overline{u_i}}{\partial x_j} \tag{4.317}$$

where $\kappa=0.41$ is the von Kármán constant. The published value for the WALE constant, C_w, is 0.5; however, intensive validation during a European Union research project involving the original model developers has shown consistently superior results in Ansys Fluent with C_w = 0.325, and so it is this value that is used as the default setting. The rest of the notation is the same as for the Smagorinsky-Lilly model. With this spatial operator, the WALE model is designed to return the correct wall asymptotic (y^3) behavior for wall bounded flows.

Another advantage of the WALE model is that it returns a zero turbulent viscosity for laminar shear flows. This allows the correct treatment of laminar zones in the domain. In contrast, the Smagorinsky-Lilly model produces nonzero turbulent viscosity. The WALE model is therefore preferable compared to the Smagorinsky-Lilly model.

4.16.2.4. Algebraic Wall-Modeled LES Model (WMLES)

While widely used in the academic community, LES has had very limited impact on industrial simulations. The reason lies in the excessively high resolution requirements for wall boundary layers. Near the wall, the largest scales in the turbulent spectrum are nevertheless geometrically very small and require a very fine grid and a small time step. In addition, unlike RANS, the grid cannot only be refined in the wall normal direction, but also must resolve turbulence in the wall parallel plane. This can only be achieved for flows at very low Reynolds number and on very small geometric scales (the extent of the LES domain cannot be much larger than 10-100 times the boundary layer thickness parallel to the wall). For this reason, using LES is only recommended for flows where wall boundary layers are not relevant and need not be resolved, or for flows where the boundary layers are laminar due to the low Reynolds number.

However, there are only very few such flows and other approaches need to be employed. A promising approach to overcome the Reynolds number scaling limitations of LES is the algebraic Wall-Modeled LES (WMLES) approach [589] (p. 1087). In WMLES, the RANS portion of the model is only activated in the inner part of the logarithmic layer and the outer part of the boundary layer is covered by a modified LES formulation. Since the inner portion of the boundary layer is responsible for the Reynolds number dependency of the LES model, the WMLES approach can be applied at the same grid resolution to an ever increasing Reynolds number for channel flow simulations.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Note that for wall boundary layers, the Reynolds number scaling is not entirely avoided, as the thickness of the boundary layer declines relative to body dimensions with increasing Reynolds number. Assuming a certain number of grid nodes per 'boundary layer volume', the overall grid spacing will decrease and the overall number of cells will increase with the Reynolds number.

Another interesting option, and possibly the best use of the WMLES option, is when it is combined with the Embedded LES (ELES) implementation in Ansys Fluent. ELES limits the LES zone to small regions of the domain and covers the less critical areas with RANS. WMLES extends this functionality to high Reynolds numbers in the LES domain. To apply this option, refer to Setting Up the Large Eddy Simulation Model and Setting Up the Embedded Large Eddy Simulation (ELES) Model in the User's Guide.

4.16.2.4.1. Algebraic WMLES Model Formulation

The original Algebraic WMLES formulation was proposed in the works of Shur et.al. [589] (p. 1087). It combines a mixing length model with a modified Smagorinsky model [600] (p. 1088) and with the wall-damping function of Piomelli [517] (p. 1083).

In the Shur et al. model [589] (p. 1087), the eddy viscosity is calculated with the use of a hybrid length scale:

$$\nu_t = \min\left[(\kappa d_w)^2, (C_{Smag}\Delta)^2\right] \cdot S \cdot \left\{1 - \exp\left[-\left(y^+/25\right)^3\right]\right\} \tag{4.318}$$

where d_w is the wall distance, S is the strain rate, $\kappa = 0.4187$, and $C_{S_{mag}} = 0.2$ are constants, and y^+ is the normal to the wall inner scaling. The LES model is based on a modified grid scale to account for the grid anisotropies in wall-modeled flows:

$$\Delta = \min\left(\max\left(C_w \cdot d_w;\, C_w \cdot h_{max}, h_{wn}\right); h_{max}\right) \tag{4.319}$$

Here, h_{max} is the maximum edge length for a rectilinear hexahedral cell (for other cell types and/or conditions an extension of this concept is used). h_{wn} is the wall-normal grid spacing, and $C_w = 0.15$ is a constant.

The above model has been optimized for use in Ansys Fluent but remains within the spirit of the original model.

4.16.2.4.1.1. Reynolds Number Scaling

The main advantage of the WMLES formulation is the improved Reynolds number scaling. The classical resolution requirements for wall resolved LES is

$$\Delta_x^+ = 40, \quad Ny = 30, \quad \Delta_z^+ = 20 \tag{4.320}$$

where

$$\Delta_x^+ = \frac{u_\tau \Delta x}{\nu}, \quad \Delta_z^+ = \frac{u_\tau \Delta z}{\nu} \tag{4.321}$$

and Ny is the number of cells across the boundary layer, h, (or half channel height h) and ν is the kinematic viscosity. The wall friction velocity is defined as

$$u_\tau = \sqrt{\frac{\tau_w}{\rho}} \tag{4.322}$$

where τ_w is the wall shear stress and ρ is the density.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

The resolution requirement for WMLES is

$$\Delta_x=\frac{h}{10}, \qquad Ny=30-40, \qquad \Delta_z=\frac{h}{20} \tag{4.323}$$

Again with $h=\frac{H}{2}$ for a channel (where *H* is the channel height), or $h=\delta$ (where δ is the boundary layer thickness).

Considering a periodic channel (in the span and streamwise direction) with the geometry dimensions:

$$Lx=8h, \qquad Ly=2h, \qquad Lz=3h \tag{4.324}$$

you obtain the following cell numbers (N_T total number of cells in channel) for the wall-resolved LES and WMLES:

Wall-Resolved LES:

Table 4.1: Wall-Resolved Grid Size as a Function of Reynolds Number

Re_τ	500	10^3	10^4	10^5
N_T	5.0e5	1.8e6	1.8e8	1.8e10

WMLES:

Table 4.2: WMLES Grid Size as a Function of Reynolds Number

Re_τ	500	10^3	10^4	10^5
N_T	5.0e5	5.0e5	5.0e5	5.0e5
Ratio LES/WMLES	1	4	$4x10^2$	$4x10^4$
Ratio CPU LES/WMLES (CFL=0.3)	1	10^1	10^4	10^7

The Reynolds number is based on the friction velocity:

$$Re_\tau=\sqrt{\frac{u_\tau h}{\nu}} \tag{4.325}$$

From the above tables, it can be seen that the quadratic Reynolds number dependence of the Wall-Resolved LES has been avoided by WMLES and that the CPU effort is substantially reduced.

However, it is still important to remember that WMLES is much more computationally expensive than RANS. Considering a wall boundary layer flow, with WMLES one must cover every boundary layer volume $(\delta\times\delta\times\delta)$ with 10 x (30-40) x 20 = 6000-8000 cells in the stream-wise, normal, and span-wise direction. RANS can be estimated to 1 x (30-40) x 1 =30-40 cells. In addition, RANS can be computed steady-state whereas WMLES requires an unsteady simulation with CFL~0.3.

4.16.2.4.2. Algebraic WMLES S-Omega Model Formulation

One of the deficiencies of the WMLES approach is that the model does not provide zero eddy-viscosity for flows with constant shear when you use a modified Smagorinsky model in the LES

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

zone. For this reason, the WMLES model does not allow the computation of transitional effects, and can produce overly large eddy-viscosities in separating shear layers (like from a backstep).

A way to enhance the WMLES formulation given in Equation 4.318 (p. 124) is to compute the LES portion of the model using the difference $\mathrm{abs}(S-\Omega)$ instead of S, where S is the strain rate and Ω is the vorticity magnitude. This enhancement is referred to as the WMLES S-Ω formulation.

4.16.2.5. Dynamic Kinetic Energy Subgrid-Scale Model

The original and dynamic Smagorinsky-Lilly models, discussed previously, are essentially algebraic models in which subgrid-scale stresses are parameterized using the resolved velocity scales. The underlying assumption is the local equilibrium between the transferred energy through the grid-filter scale and the dissipation of kinetic energy at small subgrid scales. The subgrid-scale turbulence can be better modeled by accounting for the transport of the subgrid-scale turbulence kinetic energy.

The dynamic subgrid-scale kinetic energy model in Ansys Fluent replicates the model proposed by Kim and Menon [308] (p. 1071).

The subgrid-scale kinetic energy is defined as

$$k_{sgs}=\frac{1}{2}\left(\overline{u_k^2}-\overline{u}_k^2\right) \tag{4.326}$$

which is obtained by contracting the subgrid-scale stress in Equation 4.11 (p. 43).

The subgrid-scale eddy viscosity, μ_t, is computed using k_{sgs} as

$$\mu_t=C_k\rho k_{sgs}^{1/2}\Delta_f \tag{4.327}$$

where Δ_f is the filter-size computed from $\Delta_f\equiv V^{1/3}$.

The subgrid-scale stress can then be written as

$$\tau_{ij}-\frac{2}{3}\rho k_{sgs}\delta_{ij}=-2C_k\rho k_{sgs}^{1/2}\Delta_f\overline{S}_{ij} \tag{4.328}$$

k_{sgs} is obtained by solving its transport equation

$$\rho\frac{\partial\overline{k}_{sgs}}{\partial t}+\rho\frac{\partial\overline{u}_j\overline{k}_{sgs}}{\partial x_j}=-\tau_{ij}\frac{\partial\overline{u}_i}{\partial x_j}-C_\varepsilon\rho\frac{k_{sgs}^{3/2}}{\Delta_f}+\frac{\partial}{\partial x_j}\left(\frac{\mu_t}{\sigma_k}\frac{\partial k_{sgs}}{\partial x_j}\right) \tag{4.329}$$

In the above equations, the model constants, C_k and C_ε, are determined dynamically [308] (p. 1071). σ_k is hardwired to 1.0. The details of the implementation of this model in Ansys Fluent and its validation is given by Kim [305] (p. 1070).

4.16.3. Inlet Boundary Conditions for Scale Resolving Simulations

This section describes the algorithms available in Ansys Fluent to model the fluctuating velocity at velocity inlet boundaries or pressure inlet boundaries.

The stochastic components of the flow at the velocity-specified inlet boundaries are neglected if the **No Perturbations** option is used. In such cases, individual instantaneous velocity components are simply set equal to their mean velocity counterparts. This option is suitable only when the level of

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

turbulence at the inflow boundaries is negligible or does not play a major role in the accuracy of the overall solution.

Important:

The methods listed below are available for velocity inlet and pressure inlet boundary conditions. For the velocity inlet, the fluctuations are added on the mean specified velocity. For the pressure inlet, virtual body forces are employed in the momentum equations to add the reconstructed turbulent fluctuations to the velocity field. These virtual body forces are considered only in the first cells close to the inlet.

The methods are available for the SAS, DES, SDES, SBES, and LES models. If unsteadiness inside the domain can be sustained depends on the flow and the grid spacing, especially when using the SAS or the DDES models for wall bounded flows.

These methods require realistic inlet conditions, such as providing U, k, and ε (or ω) profiles, that can be obtained from separate RANS simulations. Unrealistic ("flat") turbulent profiles at inlets will generate unrealistic turbulent eddies at inlets. Be aware of the order of the conversion operations applied to the inlet velocity and to the turbulence quantities, which is explained in Estimating Turbulent Kinetic Energy from Turbulence Intensity. This conversion order is important when computing the fluctuation amplitudes based on the specified turbulence intensity. It can also influence other properties of the generated synthetic turbulence, for example the global time scale τ of the Synthetic Turbulence Generator, see Equation 4.337 (p. 129). This time scale depends on the inlet velocity, which may first be converted according to the Velocity Formulation selected in the General Task Page.

4.16.3.1. Vortex Method

To generate a time-dependent inlet condition, a random 2D vortex method is considered. With this approach, a perturbation is added on a specified mean velocity profile via a fluctuating vorticity field (that is, two-dimensional in the plane normal to the streamwise direction). The vortex method is based on the Lagrangian form of the 2D evolution equation of the vorticity and the Biot-Savart law. A particle discretization is used to solve this equation. These particles, or "vortex points" are convected randomly and carry information about the vorticity field. If N is the number of vortex points and A is the area of the inlet section, the amount of vorticity carried by a given particle i is represented by the circulation Γ_i and an assumed spatial distribution η:

$$\Gamma_i(x,y) = 4\sqrt{\frac{\pi A k(x,y)}{3N[2\ln(3)-3\ln(2)]}} \tag{4.330}$$

$$\eta(\vec{x}) = \frac{1}{2\pi\sigma^2}\left(2e^{-|x|^2/2\sigma^2}-1\right)2e^{-|x|^2/2\sigma^2} \tag{4.331}$$

where k is the turbulence kinetic energy. The parameter σ provides control over the size of a vortex particle. The resulting discretization for the velocity field is given by

$$\vec{u}(\vec{x}) = \frac{1}{2\pi}\sum_{i=1}^{N}\Gamma_i\frac{\left((\vec{x}_i-\vec{x})\times\vec{z}\right)\left(1-e^{|\vec{x}-\vec{x}|^2/2\sigma^2}\right)}{|\vec{x}-\vec{x}_i'|^2} \tag{4.332}$$

Where $\vec{z}$ is the unit vector in the streamwise direction. Originally [585] (p. 1087), the size of the vortex was fixed by an ad hoc value of σ. To make the vortex method generally applicable, a local vortex

size is specified through a turbulent mixing length hypothesis. σ is calculated from a known profile of mean turbulence kinetic energy and mean dissipation rate at the inlet according to the following:

$$\sigma=\frac{ck^{3/2}}{2\varepsilon} \tag{4.333}$$

where $c=0.16$. To ensure that the vortex will always belong to resolved scales, the minimum value of σ in Equation 4.333 (p. 128) is bounded by the local grid size. The sign of the circulation of each vortex is changed randomly each characteristic time scale τ. In the general implementation of the vortex method, this time scale represents the time necessary for a 2D vortex convected by the bulk velocity in the boundary normal direction to travel along n times its mean characteristic 2D size (σ_m), where n is fixed equal to 100 from numerical testing. The vortex method considers only velocity fluctuations in the plane normal to the streamwise direction.

In Ansys Fluent however, a simplified linear kinematic model (LKM) for the streamwise velocity fluctuations is used [411] (p. 1076). It is derived from a linear model that mimics the influence of the two-dimensional vortex in the streamwise mean velocity field. If the mean streamwise velocity U is considered as a passive scalar, the fluctuation u' resulting from the transport of U by the planar fluctuating velocity field v' is modeled by

$$u'=-\vec{v}'\cdot\vec{g} \tag{4.334}$$

where $\vec{g}$ is the unit vector aligned with the mean velocity gradient $\overrightarrow{\nabla U}$. When this mean velocity gradient is equal to zero, a random perturbation can be considered instead.

Since the fluctuations are equally distributed among the velocity components, only the prescribed kinetic energy profile can be fulfilled at the inlet of the domain. Farther downstream, the correct fluctuation distribution is recovered [411] (p. 1076). However, if the distribution of the normal fluctuations is known or can be prescribed at the inlet, a rescaling technique can be applied to the synthetic flow field in order to fulfill the normal statistic fluctuations $<uu>$, $<vv>$, and $<ww>$ as given at the inlet.

With the rescaling procedure, the velocity fluctuations are expressed according to:

$$u_i'^{*}=u_i'\frac{\sqrt{<u_iu_i>}}{\sqrt{2/3k}} \tag{4.335}$$

This also results in an improved representation of the turbulent flow field downstream of the inlet. This rescaling procedure is used only if the **Reynolds-Stress Components** is specified as the **Reynolds-Stress Specification Method**, instead of the default option **K or Turbulence Intensity**.

Note:

Mass conservation is not enforced for the velocity fluctuations generated by the **Vortex Method**. This can lead to unexpected pressure fluctuations and artificial disturbances or noise when solving acoustics. To remedy this problem, you can enable mass conser-

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

vation for the vortex method by selecting **Satisfy Mass Conservation?** on the inlet dialog box.

Important:

Since the vortex method theory is based on the modification of the velocity field normal to the streamwise direction, it is imperative that you create an inlet plane normal (or as close as possible) to the streamwise velocity direction. Problems can arise if the inlet convective velocity is of the same order or lower than $\mathrm{k}^{0.5}$, as this can cause local outflow. For this reason it should be applied with caution for Tu~1 or smaller).

The vortex method covers the entire boundary for which it is selected with vortices, even in regions, where the incoming turbulence level is low (freestream). For efficiency reasons, it is therefore recommended that you restrict the domain of the VM to the region where you which to generate unsteady turbulence. As an example, assume a thin mixing layer enters through a much larger inlet region. It is then advised to split the inlet boundary into three parts. One below the mixing layer with constant inlet velocity, one where the mixing layer enters the domain using the VM and a third above the mixing layer, again with a constant velocity. Thereby the vortices are concentrated at the domain of interest. Otherwise, a very large number of vortices would have to be generated to ensure a sufficient number of them in the thin mixing layer region.

4.16.3.2. Spectral Synthesizer

The spectral synthesizer provides an alternative method of generating fluctuating velocity components. It is based on the random flow generation technique originally proposed by Kraichnan [320] (p. 1071) and modified by Smirnov et al. [602] (p. 1088). In this method, fluctuating velocity components are computed by synthesizing a divergence-free velocity-vector field from the summation of Fourier harmonics. In Ansys Fluent, the default number of Fourier harmonics is set to 250.

4.16.3.3. Synthetic Turbulence Generator

To generate time-dependent inlet conditions for scale resolving turbulence simulations, a Fourier based synthetic turbulence generator is used.

This method is inexpensive in terms of computational time compared with the other existing methods while achieving high quality turbulence fluctuations.

The velocity vector at a point $\mathbf{r}=\{x,y,z\}$ of an inlet boundary condition is specified as the sum of the mean velocity vector and the vector of synthetic velocity fluctuations:

$$\mathbf{U}(\mathbf{r},t)=\mathbf{U}_{Inlet,mean}(\mathbf{r})+\mathbf{u}'(\mathbf{r},t) \tag{4.336}$$

The field of velocity fluctuations $\mathbf{u}'_i(\mathbf{r},t)$ is defined as the superposition of weighted spatio-temporal Fourier modes scaled such that the corresponding kinetic energy of the inflow is equal to the prescribed RANS kinetic energy K_t at the inlet boundary:

$$\mathbf{u}'(\mathbf{r},t)=2K_t\sum_{n=1}^{N}\sqrt{q^n}\left[\boldsymbol{\sigma}^n\cos\left(k^n\mathbf{d}^n\cdot\mathbf{r}+\phi^n+s^n\frac{t}{\tau}\right)\right] \tag{4.337}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

where:

N is the number of modes.

$q^n$is the normalized amplitude of the mode n defined by the local energy spectrum.

k^n is the amplitude of the wave number vector of the mode n.

$\mathbf{d}^n$ is a random unit vector of direction uniformly distributed over a sphere.

$\boldsymbol{\sigma}^n$ is a unit vector normal to $\mathbf{d}^n$($\boldsymbol{\sigma}^n \cdot \mathbf{d}^n = 0$), and in turn the angle defining its direction in the plane normal to $\mathbf{d}^n$ is a random number uniformly distributed in the interval $(0,2\pi)$.

φ^n is the phase of the mode n, which is also a random number uniformly distributed in the interval $(0,2\pi)$.

s^n is the non-dimensional frequency of the mode n with Gaussian distribution and the mean value and standard deviation equal to 2π.

τ is the global time-scale. It is based on the characteristic length and velocity scales defined as the maximum RANS turbulence length scale and velocity at the inlet zone.

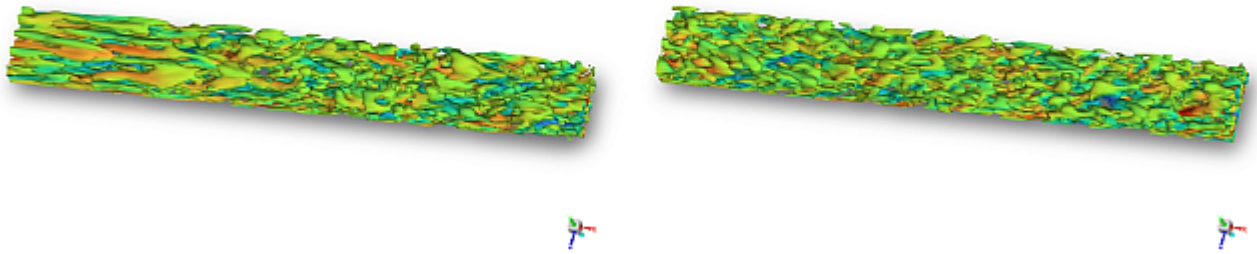
Important:

Using the entire inlet zone to determine the characteristic scales may lead to incorrect results if only a small section of the inlet face zone generates synthetic turbulence. For example, unrealistically long turbulent structures can be generated in the turbulent boundary layer if the RANS turbulent length scale in the free stream is arbitrarily high. In such situations, it is necessary to limit the area of computation for the global scales to the relevant part of the inlet only, the effect of which can be seen in the figure below. To limit the search area, the **Scale Search Limiters** can be specified: **Turbulent Intensity**, **Turbulent Viscosity Ratio**, or **Wall Distance Threshold**. Velocity and length scales are then computed using only the part of inlet face zone where the corresponding inlet turbulent parameter is higher than the specified threshold values (**Turbulent Intensity**, **Turbulent Viscosity Ratio**) or simply where the distance to the wall is smaller than the specified **Wall Distance Threshold** value. Limiters can be specified on the Velocity or Pressure Inlet dialog boxes or using the console command to set boundary conditions (`/define/boundary-conditions/set`).

Figure 4.11: Turbulent Structures Generated by Synthetic Turbulence Generator in the Boundary Layer with Arbitrarily High Turbulent Length Scale in the Free Stream:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

With Scale Search Limiter Option Set to None (left) and with Specified Turbulent Intensity Threshold 2% (right)



The face based (default) synthetic turbulence generator generates spurious pressure fluctuations making it unfit for aeroacoustics simulations. The volumetric version, available only for the omega-based SBES models, can be enabled for velocity or pressure inlets by selecting **Volumetric Forcing?**. It avoids generating undesirable noise by introducing volumetric sources of turbulent fluctuations into the momentum equations within a three dimensional forcing zone starting from the boundary. The reduction in noise allows the synthetic turbulence generator to be applied to acoustic tasks.

Inside the volumetric forcing region, the SBES model works in LES mode and that can cause inaccuracies in the presence of boundary layers (for example, if the volumetric synthetic turbulence generation is applied to a boundary condition that is connected to a wall). The SBES model behavior can be checked by visualizing the SBES blending function.

You have the option of specifying an **Automatic** (based on turbulence length scale) forcing zone thickness or directly setting the forcing zone thickness via a **Specified Value**.

Note:

Unlike the **Vortex Method**, mass conservation for the **Synthetic Turbulence Generator** is enabled by default and cannot be disabled.

For further details, see [591] (p. 1087).

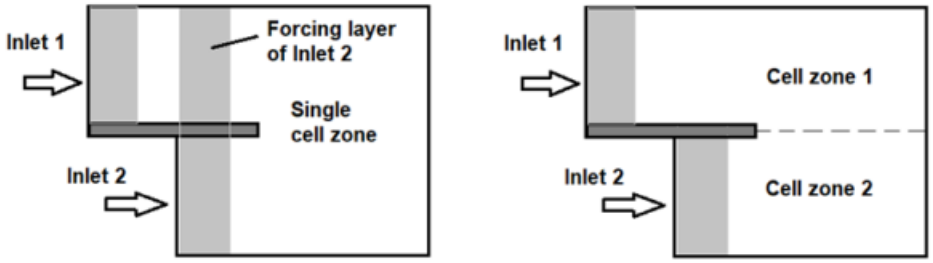
4.16.3.3.1. Limitations

A universal implementation of the volumetric forcing is a challenging task, since it requires a correct identification of the forcing zone. In the current version of this method in Ansys Fluent, all cells in a cell zone adjacent to the considered inlet zone are marked for the forcing, which are located within the forcing zone thickness counted from the inlet zone. The distance between each cell center and the inlet zone center is calculated along the normal-to-inlet direction. In the parallel-to-inlet direction the marked cell layer may be larger that the inlet zone. However, the forcing intensity is scaled with the modeled turbulence kinetic energy, which results from its transport equation. Therefore, outside of the turbulent stream coming from the adjacent inlet zone, the forcing intensity is zero. When, however, the marked cell layer crosses other turbulent parts of the solution field, this will generate forcing also in those parts, which is normally undesir-

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

able. Such situation is illustrated in Figure 4.12: Splitting of a cell zone in a complex flow simulation to avoid the undesirable side effect of the volumetric forcing (p. 132)Figure 4.12.

Figure 4.12: Splitting of a cell zone in a complex flow simulation to avoid the undesirable side effect of the volumetric forcing



In such cases splitting a large cell zone in smaller parts helps to limit the marked cell layer and to avoid spurious forcing in undesired locations.

4.17. Embedded Large Eddy Simulation (ELES)

This section describes the theory behind the Embedded Large Eddy Simulation (ELES) model. Information is presented in the following sections:

4.17.1. Overview

4.17.2. Selecting a Model

4.17.3. Interfaces Treatment

4.17.1. Overview

Large Eddy Simulation (LES) has had very limited impact on industrial CFD simulations, mainly due to its high computational costs. There are only very few technical applications, where LES can be applied within the entire computational domain. Such flows are typically of very low Reynolds number, or flows where wall boundary layers are not important (free shear flows). Especially the high resolution requirements for wall bounded flows even at moderate Reynolds numbers have severely limited the usage of LES.

In order to allow the resolution of large turbulent structures in industrial flow simulations, hybrid models like Scale-Adaptive Simulation (SAS) (see Scale-Adaptive Simulation (SAS) Model (p. 106)), Detached Eddy Simulation (DES) (see Detached Eddy Simulation (DES) (p. 109)), Shielded Detached Eddy Simulation (SDES) (Shielded Detached Eddy Simulation (SDES) (p. 114)), and Stress-Blended Eddy Simulation (SBES) (Stress-Blended Eddy Simulation (SBES) (p. 117)) have been developed. For these models, the wall boundary layers are typically covered by the RANS part of the model and turbulence is only resolved in large separated (detached) zones. The unsteadiness in the simulations is generated from a global flow instability as observed behind bluff bodies. However, such an approach is not always suitable, as not all flows exhibit a sufficiently strong instability to generate turbulent structures by themselves. In such situations, zonal models are desirable, where a clear distinction between RANS and LES regions can be made and where turbulence is converted from RANS to LES by suitable methods at the interface. One such approach is ELES, where you generate RANS and LES zones during the grid generation phase, select appropriate models for each zone, and define the appropriate

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

treatment at the interface. ELES is therefore not a new turbulence model, but the combination of RANS and LES models joined by appropriate interface conditions.

4.17.2. Selecting a Model

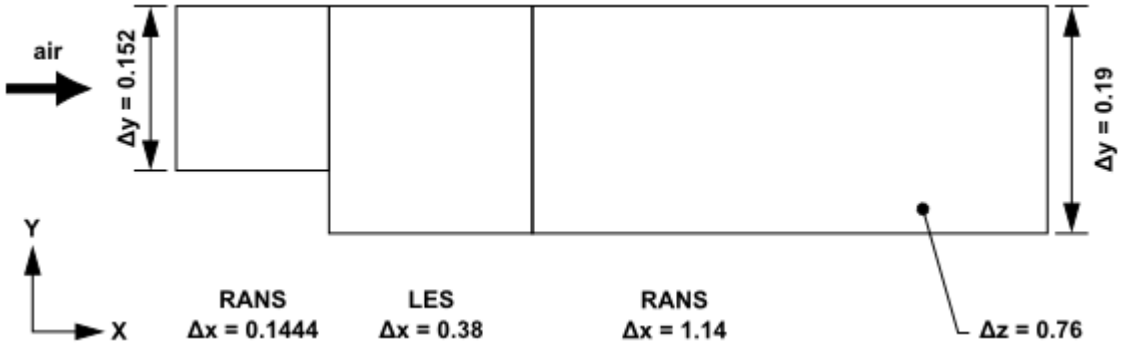
In principle, there are a large number of models that can be combined. Generally, all RANS models in Ansys Fluent can be selected in the RANS region. The RANS model not compatible with ELES is the Spalart-Allmaras model, as a one-equation model cannot provide the required turbulent length scale to the interface method.

With respect to LES models, all algebraic LES models are available (this excludes the Dynamic Kinetic Energy Subgrid-Scale Model — see Dynamic Kinetic Energy Subgrid-Scale Model (p. 126)). While for the RANS portion, a general recommendation is difficult due to the different strength of RANS models for different applications, for the LES region, the WALE model is generally considered a suitable choice.

4.17.3. Interfaces Treatment

Figure 4.13: Backward Facing Step Flow Using ELES (p. 133) shows a typical example of an ELES scenario. The geometry of the simulation is divided into RANS and LES zones with interfaces between them. The upstream and the downstream zone are covered by RANS and the middle section with the reversed flow is covered by LES.

Figure 4.13: Backward Facing Step Flow Using ELES



As the flow passes between the RANS domain and the LES domain, and from the LES domain into the downstream RANS domain, the treatment of the interfaces must be considered, as described in the following sections.

4.17.3.1. RANS-LES Interface

4.17.3.2. LES-RANS Interface

4.17.3.3. Internal Interface Without LES Zone

4.17.3.4. Grid Generation Guidelines

4.17.3.1. RANS-LES Interface

The most critical interface is the interface where the flow leaves the RANS domain and enters the LES region (RANS-LES interface). At this interface, it is necessary to convert modeled turbulence kinetic energy into resolved energy and you must select an appropriate method for this transfer. This can be achieved by selecting one of the two existing methods for generating resolved turbulence at inlets – namely the Vortex Method (VM, see Vortex Method (p. 127)) or the Spectral Synthesizer

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

(SS, see Spectral Synthesizer (p. 129)). Both methods have been augmented so that they can be used at interior interfaces between the RANS and LES zone. The RANS information is obtained from the RANS model used upstream of the interface. For details and recommendations on these methods, see Zonal Modeling and Embedded LES (ELES) and Setting Up the Embedded Large Eddy Simulation (ELES) Model in the User's Guide.

If a DDES, SAS, SDES, or SBES model is run in the entire domain, the interface treatment ensures that the turbulent kinetic energy that is converted from RANS to LES is subtracted from the RANS model. Thereby avoiding double-accounting of turbulence.

4.17.3.2. LES-RANS Interface

There is also an ambiguity concerning how to revert back from the LES to the RANS zone (LES-RANS interface). The first option is to freeze the background RANS model during the ELES simulation in the LES zone. This assumes that the ELES is started from a reasonably converged RANS simulation (this is recommended in any case). The momentum equations in the LES zone are not affected by the freeze as they are provided by the eddy-viscosity from the LES model. At the 'downstream' LES-RANS interface, the RANS model is then simply reactivated, using the frozen RANS solution inside the LES zone as an 'inlet' condition for the downstream RANS zone. This works well, as long as there is no significant change in mean flow topology between the full RANS and the ELES run, or if the flow downstream of the LES-RANS interface is not relevant to the goals of the simulation. It should be stressed, however, that for this option, any non-converged RANS starting solution will affect the flow for all times downstream of the LES-RANS interface. The option of freezing the turbulence variables of the underlying RANS model inside the LES domain is employed with all standard RANS models.

A second option would be to run the RANS model in passive mode within the LES zone, meaning the model is solved on the unsteady velocity field, but the resulting eddy-viscosity is overwritten by the LES model eddy-viscosity for the momentum equations. The RANS model is then reactivated at the LES-RANS interface. It turns out that this approach fails for standard RANS models, as they severely over-predict the turbulent kinetic energy in the LES zone. The reason lies in the high strain rates computed from the unsteady velocity field. These strain rates enter into the production term Pk and generate very high turbulence levels in the LES zone. However, this option is sensible in combination with the SST-SAS model. This model recognizes the resolved scale and adjusts turbulence levels accordingly. When reactivated at the downstream interface, the model gradually converts back to RANS as the grid coarsens, or stays in LES-mode if time step and grid resolution permits.

4.17.3.3. Internal Interface Without LES Zone

There is a final option, where only a single hybrid model is run in the entire domain, that is, there is no change in the model between the RANS and LES zone. The choice is between the (rke- or SST-) DDES, the (SST-) SAS model, the (SST-) SDES model, and the (SST-) SBES model, all of which can be run in RANS and LES mode. At the interface, again, the Vortex Method or the Spectral Synthesizer can be selected to generate resolved turbulence. When using the (rke- or SST-) DDES model, one has to consider that the model will not be in RANS mode for free shear flows (flows away from walls where the DDES shielding function is not active) if the grid is sufficiently fine to activate the DES limiter in the RANS zone. The combination of the (I)DDES model with the interface option is therefore mostly limited to wall-bounded internal flows. For more information about DDES, see Detached Eddy Simulation (DES) (p. 109), and for more information about IDDES, see Improved Delayed Detached Eddy Simulation (IDDES) (p. 113).

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

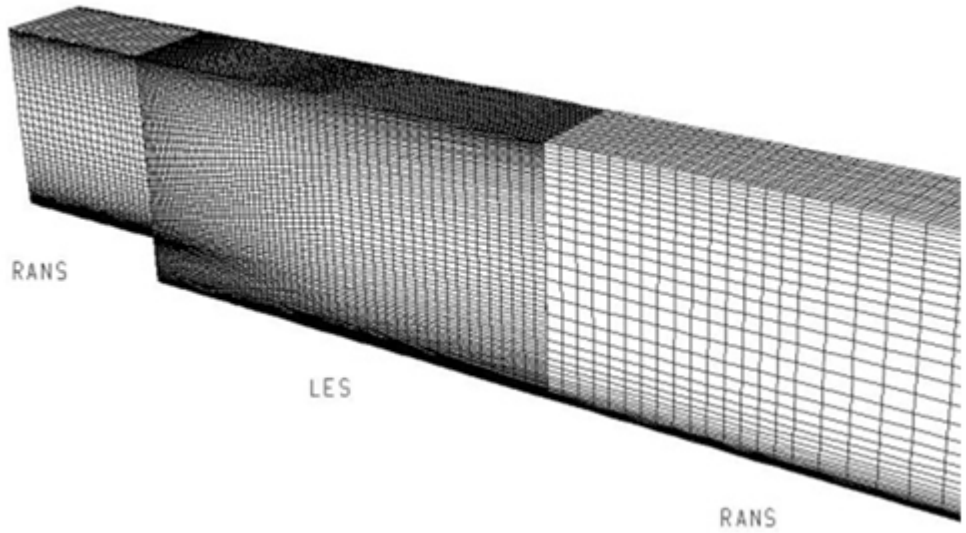
The current option of using an internal interface works more naturally in combination with the SST-SAS model, where little attention is required to carry the model in RANS mode to the interface. The current interface treatment can therefore be used in situations, where the internal flow instability is not strong enough to convert the SAS model into scale-resolving ('LES') mode at a desired location. Downstream of the interface, the SAS model will then remain in LES mode if the grid and time step size permit, or will gradually turn back to RANS otherwise.

If a DDES, SAS, SDES, or SBES model is run in the entire domain, the interface treatment ensures that the turbulent kinetic energy that is converted from RANS to LES is subtracted from the RANS model. This avoids a double-accounting of the turbulence.

4.17.3.4. Grid Generation Guidelines

The grids used in the RANS and LES zones have to be conforming to the resolution requirements of the underlying turbulence model. Figure 4.14: Typical Grid for ELES for Backward Facing Step (p. 135) shows an example of an ELES grid. Frequently, the LES zone will be connected to the RANS zones by Non-Conformal Interfaces (NCI) to allow a more refined grid in the LES zone.

Figure 4.14: Typical Grid for ELES for Backward Facing Step



4.18. Near-Wall Treatments for Wall-Bounded Turbulent Flows

Information about near-wall treatments for wall-bounded turbulent flows is presented in the following sections:

4.18.1. Overview
4.18.2. Wall Treatment for ε-based Models
4.18.3. y+-Insensitive Near-Wall Treatment for ω-based Turbulence Models
4.18.4. LES Near-Wall Treatment
4.18.5. Wall Roughness Effects in Turbulent Wall-Bounded Flows

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

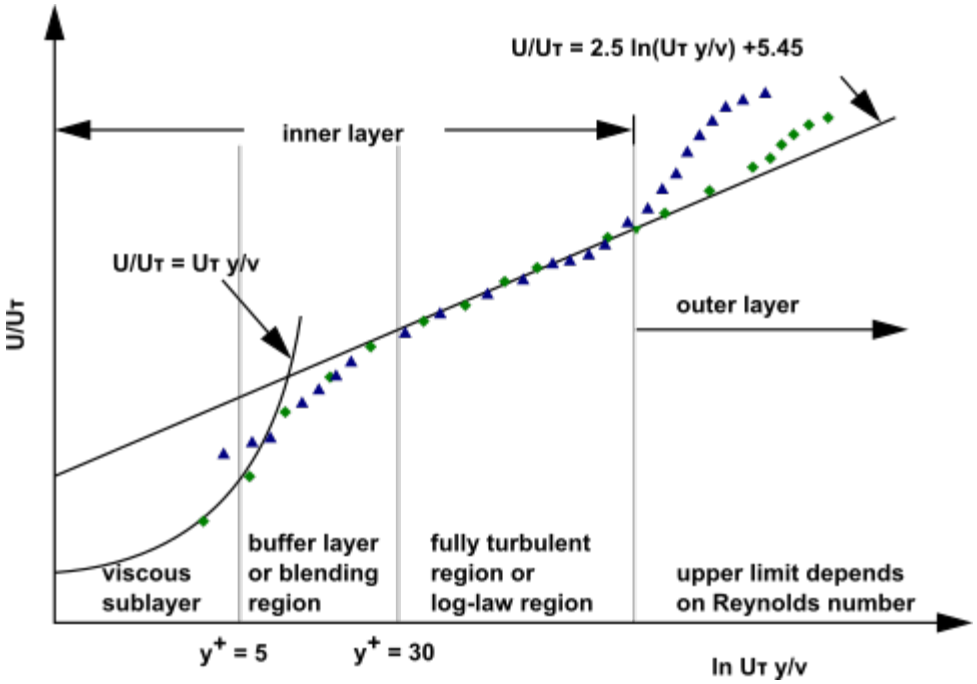
4.18.1. Overview

Turbulent flows are significantly affected by the presence of walls. Obviously, the mean velocity field is affected through the no-slip condition that has to be satisfied at the wall. However, the turbulence is also changed by the presence of the wall in non-trivial ways. Very close to the wall, viscous damping reduces the tangential velocity fluctuations, while kinematic blocking reduces the normal fluctuations. Toward the outer part of the near-wall region, however, the turbulence is rapidly augmented by the production of turbulence kinetic energy due to the large gradients in mean velocity.

The near-wall modeling significantly impacts the fidelity of numerical solutions, inasmuch as walls are the main source of mean vorticity and turbulence. After all, it is in the near-wall region that the solution variables have large gradients, and the momentum and other scalar transports occur most vigorously. Therefore, accurate representation of the flow in the near-wall region determines successful predictions of wall-bounded turbulent flows.

Numerous experiments have shown that the near-wall region can be largely subdivided into three layers. In the innermost layer, called the "viscous sublayer", the flow is almost laminar, and the (molecular) viscosity plays a dominant role in momentum and heat or mass transfer. In the outer layer, called the fully-turbulent layer, turbulence plays a major role. Finally, there is an interim region between the viscous sublayer and the fully turbulent layer where the effects of molecular viscosity and turbulence are equally important. Figure 4.15: Subdivisions of the Near-Wall Region (p. 136) illustrates these subdivisions of the near-wall region, plotted in semi-log coordinates.

Figure 4.15: Subdivisions of the Near-Wall Region



In Figure 4.15: Subdivisions of the Near-Wall Region (p. 136), $y^+ \equiv \rho u_\tau y / \mu$, where u_τ is the friction velocity, defined as $\sqrt{\frac{\tau_w}{\rho}}$.

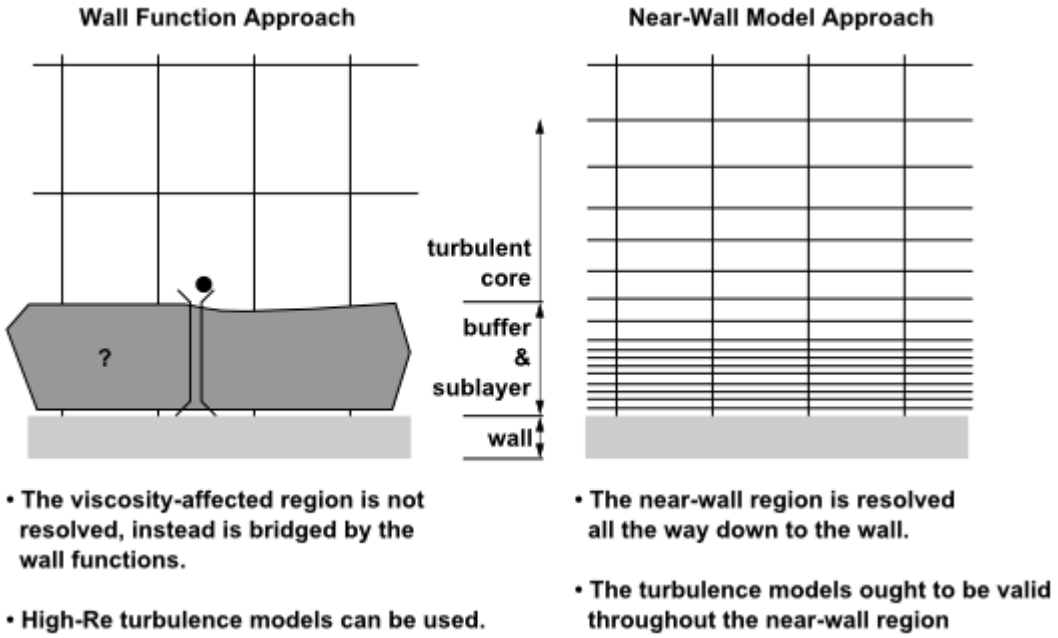
Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Wall Functions vs. Near-Wall Model

Traditionally, there are two approaches to modeling the near-wall region. In one approach, the viscosity-affected inner region (viscous sublayer and buffer layer) is not resolved. Instead, semi-empirical formulas called "wall functions" are used to bridge the viscosity-affected region between the wall and the fully-turbulent region. The use of wall functions obviates the need to modify the turbulence models to account for the presence of the wall.

In another approach, the turbulence models are modified to enable the viscosity-affected region to be resolved with a mesh all the way to the wall, including the viscous sublayer. For the purposes of discussion, this will be termed the "near-wall modeling" approach. These two approaches are depicted schematically in Figure 4.16: Near-Wall Treatments in Ansys Fluent (p. 137).

Figure 4.16: Near-Wall Treatments in Ansys Fluent



The main shortcoming of all wall functions (except the scalable wall function) is that the numerical results deteriorate under refinement of the grid in the wall normal direction. y^+ values less than 15 will gradually result in unbounded errors in wall shear stress and wall heat transfer. While this was the industrial standard some years ago, Ansys Fluent has taken steps to offer more advanced wall formulations, which allow a consistent mesh refinement without a deterioration of the results. Such y^+-independent formulations are the default for all ω-equation-based turbulence models. For the ε-equation-based models, the Menter-Lechner and Enhanced Wall Treatment (EWT) serve the same purpose. A y^+-insensitive wall treatment is also the default for the Spalart-Allmaras model and allows you to run this model independent of the near-wall y^+ resolution.

High quality numerical results for the wall boundary layer will only be obtained if the overall resolution of the boundary layer is sufficient. This requirement is actually more important than achieving certain y^+ values. The minimum number of cells to cover a boundary layer accurately is around 10, but values of 20 are desirable. It should also be noted that an improvement of boundary layer resolution can often be achieved with moderate increase in numerical effort, as it requires only a mesh refinement in the wall-normal direction. The associated increase in accuracy is typically well worth the additional computing costs. For unstructured meshes, it is recommended that you generate prism layers near

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

the wall with 10–20 or more layers for an accurate prediction of the wall boundary layers. The thickness of the prism layer should be designed to ensure that around 15 or more nodes are actually covering the boundary layer. This can be checked after a solution is obtained, by looking at the turbulent viscosity, which has a maximum in the middle of the boundary layer – this maximum gives an indication of the thickness of the boundary layer (twice the location of the maximum gives the boundary layer edge). It is essential that the prism layer is thicker than the boundary layer as otherwise there is a danger that the prism layer confines the growth of the boundary layer.

Recommendations:

- For turbulence models based on the ε-equation:
  - Use Menter-Lechner (ML-ε) or Enhanced Wall Treatment (EWT-ε).
  - If wall functions are favored, use Scalable Wall Functions.
- For turbulence models based on the ω-equation:
  - The default "correlation" already offers a well calibrated y^+-insensitive wall treatment.
  - The "tabulated" y^+-insensitive wall treatment provides further improvement for BSL, SST, and GEKO.
- For the Spalart-Allmaras model, use the default y^+-insensitive wall treatment.
- For LES, use the Harmonic Blending Wall Functions Based on r+ (p. 157).

4.18.2. Wall Treatment for ε-based Models

For the ε-based models, the following wall functions are available:

4.18.2.1. Standard Wall Functions
4.18.2.2. Scalable Wall Functions
4.18.2.3. Non-Equilibrium Wall Functions
4.18.2.4. Enhanced Wall Treatment ε-Equation (EWT-ε)
4.18.2.5. Menter-Lechner ε-Equation (ML-ε)
4.18.2.6. User-Defined Wall Functions

4.18.2.1. Standard Wall Functions

The standard wall functions in Ansys Fluent are based on the work of Launder and Spalding [341] (p. 1072), and have been most widely used in industrial flows. They are provided as a default option in Ansys Fluent.

4.18.2.1.1. Momentum

The law-of-the-wall for mean velocity yields

$$U^*=\frac{1}{\kappa}\ln\left(Ey^*\right) \tag{4.338}$$

where

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$U^* \equiv \frac{U_P C_\mu^{1/4} k_P^{1/2}}{\tau_w / \rho} \tag{4.339}$$

is the dimensionless velocity.

$$y^* \equiv \frac{\rho C_\mu^{1/4} k_P^{1/2} y_P}{\mu} \tag{4.340}$$

is the dimensionless distance from the wall.

and

κ = von Kármán constant (= 0.4187)

E = empirical constant (= 9.793)

U_P = mean velocity of the fluid at the wall-adjacent cell centroid, P

k_P = turbulence kinetic energy at the wall-adjacent cell centroid, P

y_P = distance from the centroid of the wall-adjacent cell to the wall, P

μ = dynamic viscosity of the fluid

The range of y^* values for which wall functions are suitable depend on the overall Reynolds number of the flow. The lower limit always lies in the order of y^*~15. Below this limit, wall functions will typically deteriorate and the accuracy of the solutions cannot be maintained (for exceptions, see Scalable Wall Functions (p. 143)). The upper limit depends strongly on the Reynolds number. For very high Reynolds numbers (for example, ships, airplanes), the logarithmic layer can extend to values as high as several thousand, whereas for low Reynolds number flows (for example, turbine blades, and so on.) the upper limit can be as small as 100. For these low Reynolds number flows, the entire boundary layer is frequently only of the order of a few hundred y^* units. The application of wall functions for such flows should therefore be avoided as they limit the overall number of nodes one can sensibly place in the boundary layer. In general, it is more important to ensure that the boundary layer is covered with a sufficient number of (structured) cells than to ensure a certain y^* value.

In Ansys Fluent, the log-law is employed when $y^* > 11.225$. When the mesh is such that $y^* < 11.225$ at the wall-adjacent cells, Ansys Fluent applies the laminar stress-strain relationship that can be written as

$$U^* = y^* \tag{4.341}$$

It should be noted that, in Ansys Fluent, the law-of-the-wall for mean velocity and temperature are based on the wall unit, y^*, rather than y^+ ($\equiv \rho u_\tau y / \mu$). These quantities are approximately equal in equilibrium turbulent boundary layers.

4.18.2.1.2. Energy

Reynolds' analogy between momentum and energy transport gives a similar logarithmic law for mean temperature. As in the law-of-the-wall for mean velocity, the law-of-the-wall for temperature employed in Ansys Fluent is composed of the following two different laws:

- linear law for the thermal conduction sublayer, or thermal viscous sublayer, where conduction is important

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

- logarithmic law for the turbulent region where effects of turbulence dominate conduction

The thickness of the thermal conduction layer is, in general, different from the thickness of the (momentum) viscous sublayer, and changes from fluid to fluid. For example, the thickness of the thermal sublayer for a high-Prandtl-number fluid (for example, oil) is much less than its momentum sublayer thickness. For fluids of low Prandtl numbers (for example, liquid metal), on the contrary, it is much larger than the momentum sublayer thickness.

$$\frac{\delta}{\delta_T} \approx \sqrt{Pr} \tag{4.342}$$

In highly compressible flows, the temperature distribution in the near-wall region can be significantly different from that of low subsonic flows, due to the heating by viscous dissipation. In Ansys Fluent, the temperature wall functions include the contribution from the viscous heating [670] (p. 1092).

The law-of-the-wall is implemented in Ansys Fluent for the non-dimensional temperature using the wall scaling:

$$T^* \equiv \frac{\left(T_w - T_p\right)\rho c_p u^*}{\dot{q}} = T_c^* + \frac{D}{\dot{q}} \quad , \quad u^* = C_\mu^{1/4} k_p^{1/2} \tag{4.343}$$

Here the convective-conductive part T_c^* and viscous heating part D are modeled using the following composite forms:

$$T_c^* = \begin{cases} \mathrm{Pr} y^* & y^* < y_T^* \\ \mathrm{Pr}_t\left[\frac{1}{\kappa}\ln\left(E y^*\right) + P\right] & y^* > y_T^* \end{cases} \tag{4.344}$$

$$D = \begin{cases} \rho u^* \cdot \frac{1}{2}\mathrm{Pr} U_p^2 & y^* < y_T^* \\ \rho u^* \cdot \frac{1}{2}\left[\mathrm{Pr}_t U_p^2 + \left(\mathrm{Pr} - \mathrm{Pr}_t\right) U_c^2\right] & y^* > y_T^* \end{cases} \tag{4.345}$$

where P is computed by using the formula given by Jayatilleke [280] (p. 1069):

$$P = 9.24\left[\left(\frac{Pr}{Pr_t}\right)^{3/4} - 1\right]\left[1 + 0.28 e^{-0.007 Pr/Pr_t}\right] \tag{4.346}$$

and

k_P = turbulent kinetic energy at the wall-adjacent cell centroid, P

ρ = density of fluid

c_p = specific heat of fluid

$\dot{q}$ = wall heat flux

T_P = temperature at the wall-adjacent cell centroid, P

T_w = temperature at the wall

Pr = molecular Prandtl number ($\mu c_p / k_f$)

Pr_t = turbulent Prandtl number (0.85 at the wall)

A = Van Driest constant (= 26)

U_c = mean velocity magnitude at $y^* = y_T^*$

Note that, for the pressure-based solver, the terms

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$\frac{1}{2}\rho Pr\frac{C_\mu^{1/4}k_P^{1/2}}{\dot{q}}U_P^2$$

and

$$\frac{1}{2}\rho\frac{C_\mu^{1/4}k_P^{1/2}}{\dot{q}}\left\{Pr_tU_P^2+\left(Pr-Pr_t\right)U_c^2\right\}$$

will be included in Equation 4.343 (p. 140) only for compressible flow calculations.

The non-dimensional thermal sublayer thickness, y_T^*, in Equation 4.343 (p. 140) is computed as the y^* value at which the linear law and the logarithmic law intersect, given the molecular Prandtl number of the fluid being modeled.

The procedure of applying the law-of-the-wall for temperature is as follows. Once the physical properties of the fluid being modeled are specified, its molecular Prandtl number is computed. Then, given the molecular Prandtl number, the thermal sublayer thickness, y_T^*, is computed from the intersection of the linear and logarithmic profiles, and stored.

During the iteration, depending on the y^* value at the near-wall cell, either the linear or the logarithmic profile in Equation 4.343 (p. 140) is applied to compute the wall temperature T_w or heat flux $\dot{q}$ (depending on the type of the thermal boundary conditions).

The function for P given by Equation 4.346 (p. 140) is relevant for the smooth walls. For the rough walls, however, this function is modified as follows:

$$P_{rough}=3.15Pr^{0.695}\left(\frac{1}{E'}-\frac{1}{E}\right)^{0.359}+\left(\frac{E'}{E}\right)^{0.6}P \tag{4.347}$$

where E' is the wall function constant modified for the rough walls, defined by $E'=E/f_r$. To find a description of the roughness function f_r, you may refer to Equation 4.413 in the *Fluent Theory Guide* (p. 158).

4.18.2.1.3. Species

When using wall functions for species transport, Ansys Fluent assumes that species transport behaves analogously to heat transfer. Similarly to Equation 4.343 (p. 140), the law-of-the-wall for species can be expressed for constant property flow with no viscous dissipation as

$$Y^*\equiv\frac{\left(Y_{i,w}-Y_i\right)\rho C_\mu^{1/4}k_P^{1/2}}{J_{i,w}}=\begin{cases}Scy^* & \left(y^*<y_c^*\right)\\ Sc_t\left[\frac{1}{\kappa}\ln\left(Ey^*\right)+P_c\right] & \left(y^*>y_c^*\right)\end{cases} \tag{4.348}$$

where Y_i is the local species mass fraction, Sc and Sc_t are molecular and turbulent Schmidt numbers, and $J_{i,w}$ is the diffusion flux of species i at the wall. Note that P_c and y_c^* are calculated in a similar way as P and y_T^*, with the difference being that the Prandtl numbers are always replaced by the corresponding Schmidt numbers.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

4.18.2.1.4. Turbulence

In the k-ε models and in the RSM (if the option to obtain wall boundary conditions from the k equation is enabled), the k equation is solved in the whole domain including the wall-adjacent cells. The boundary condition for k imposed at the wall is

$$\frac{\partial k}{\partial n}=0 \tag{4.349}$$

where n is the local coordinate normal to the wall.

The production of kinetic energy, G_k, and its dissipation rate, ε, at the wall-adjacent cells, which are the source terms in the k equation, are computed on the basis of the local equilibrium hypothesis. Under this assumption, the production of k and its dissipation rate are assumed to be equal in the wall-adjacent control volume.

Thus, the production of k is based on the logarithmic law and is computed from

$$G_k \approx \tau_w \frac{\partial U}{\partial y}=\tau_w \frac{\tau_w}{\kappa \rho C_\mu^{1/4} k_P^{1/2} y_P} \tag{4.350}$$

and ε is computed from

$$\varepsilon_P=\frac{C_\mu^{3/4} k_P^{3/2}}{\kappa y_P} \tag{4.351}$$

The ε equation is not solved at the wall-adjacent cells, but instead is computed using Equation 4.351 (p. 142). ω and Reynolds stress equations are solved as detailed in y+-Insensitive Near-Wall Treatment for ω-based Turbulence Models (p. 152) and Wall Boundary Conditions (p. 105), respectively.

Note that, as shown here, the wall boundary conditions for the solution variables, including mean velocity, temperature, species concentration, k, and ε, are all taken care of by the wall functions. Therefore, you do not need to be concerned about the boundary conditions at the walls.

The standard wall functions described so far are provided as a default option in Ansys Fluent. The standard wall functions work reasonably well for a broad range of wall-bounded flows. However, they tend to become less reliable when the flow situations depart from the ideal conditions that are assumed in their derivation. Among others, the constant-shear and local equilibrium assumptions are the ones that most restrict the universality of the standard wall functions. Accordingly, when the near-wall flows are subjected to severe pressure gradients, and when the flows are in strong non-equilibrium, the quality of the predictions is likely to be compromised.

The non-equilibrium wall functions are offered as an additional option, which can potentially improve the results in such situations.

Important:

Standard wall functions are available with the following viscous models:

- k-ε models
- Reynolds Stress models

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

4.18.2.2. Scalable Wall Functions

Scalable wall functions avoid the deterioration of standard wall functions under grid refinement below $y^*<11$. These wall functions produce consistent results for grids of arbitrary refinement. For grids that are coarser than $y^*>11$, the standard wall functions are identical.

The purpose of scalable wall functions is to force the usage of the log law in conjunction with the standard wall functions approach. This is achieved by introducing a limiter in the y* calculations such that

$$\tilde{y}^*=MAX\left(y*,y*_{limit}\right) \tag{4.352}$$

where $y*_{limit}=11.225$. The use of Equation 4.352 (p. 143) in the context of the scalable wall functions concept is straightforward, that is, the y* formulation used for any standard wall function formula is replaced by $\tilde{y}^*$.

Scalable wall functions can be enabled in the **Viscous Model** dialog box under **Near-Wall Treatment**.

4.18.2.3. Non-Equilibrium Wall Functions

In addition to the standard wall function described above (which is the default near-wall treatment) a two-layer-based, non-equilibrium wall function [306] (p. 1070) is also available. The key elements in the non-equilibrium wall functions are as follows:

- Launder and Spalding's log-law for mean velocity is sensitized to pressure-gradient effects.
- The two-layer-based concept is adopted to compute the budget of turbulence kinetic energy ($\overline{G_k}$, $\overline{\varepsilon}$) in the wall-neighboring cells.

The law-of-the-wall for mean temperature or species mass fraction remains the same as in the standard wall functions described above.

The log-law for mean velocity sensitized to the pressure gradients is

$$\frac{\tilde{U}C_\mu^{1/4}k^{1/2}}{\tau_w/\rho}=\frac{1}{\kappa}\ln\left(E\frac{\rho C_\mu^{1/4}k^{1/2}y}{\mu}\right) \tag{4.353}$$

where

$$\tilde{U}=U-\frac{1}{2}\frac{dp}{dx}\left[\frac{y_v}{\rho\kappa\sqrt{k}}\ln\left(\frac{y}{y_v}\right)+\frac{y-y_v}{\rho\kappa\sqrt{k}}+\frac{y_v^2}{\mu}\right] \tag{4.354}$$

and y_v is the physical viscous sublayer thickness, and is computed from

$$y_v\equiv\frac{\mu y_v^*}{\rho C_\mu^{1/4}k_P^{1/2}} \tag{4.355}$$

where $y_v^*=11.225$.

The non-equilibrium wall function employs the two-layer concept in computing the budget of turbulence kinetic energy at the wall-adjacent cells, which is needed to solve the k equation at the wall-neighboring cells. The wall-neighboring cells are assumed to consist of a viscous sublayer and a fully turbulent layer. The following profile assumptions for turbulence quantities are made:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$\tau_t=\begin{cases}0, & y<y_v\\ \tau_w, & y>y_v\end{cases}\quad k=\begin{cases}\left(\frac{y}{y_v}\right)^2 k_P, & y<y_v\\ k_P, & y>y_v\end{cases}\quad \varepsilon=\begin{cases}\frac{2\nu k}{y^2}, & y<y_v\\ \frac{k^{3/2}}{C_\ell^* y}, & y>y_v\end{cases} \tag{4.356}$$

where $C_\ell^*=\kappa C_\mu^{3/4}$, and y_v is the dimensional thickness of the viscous sublayer, defined in Equation 4.355 (p. 143).

Using these profiles, the cell-averaged production of k, $\overline{G_k}$, and the cell-averaged dissipation rate, $\overline{\varepsilon}$, can be computed from the volume average of G_k and ε of the wall-adjacent cells. For quadrilateral and hexahedral cells for which the volume average can be approximated with a depth-average,

$$\overline{G_k}\equiv\frac{1}{y_n}\int_0^{y_n}\tau_t\frac{\partial U}{\partial y}dy=\frac{1}{\kappa y_n}\frac{\tau_w^2}{\rho C_\mu^{1/4}k_P^{1/2}}\ln\left(\frac{y_n}{y_v}\right) \tag{4.357}$$

and

$$\overline{\varepsilon}\equiv\frac{1}{y_n}\int_0^{y_n}\varepsilon dy=\frac{1}{y_n}\left[\frac{2\nu}{y_v}+\frac{k_P^{1/2}}{C_\ell^*}\ln\left(\frac{y_n}{y_v}\right)\right]k_P \tag{4.358}$$

where y_n is the height of the cell ($y_n=2y_P$). For cells with other shapes (for example, triangular and tetrahedral grids), the appropriate volume averages are used.

In Equation 4.357 (p. 144) and Equation 4.358 (p. 144), the turbulence kinetic energy budget for the wall-neighboring cells is effectively depends on the proportions of the viscous sublayer and the fully turbulent layer, which varies widely from cell to cell in highly non-equilibrium flows. The nonequilibrium wall functions account for the effect of pressure gradients on the distortion of the velocity profiles. In such cases the assumption of local equilibrium, when the production of the turbulent kinetic energy is equal to the rate of its destruction, is no longer valid. Therefore, the non-equilibrium wall functions, in effect, partly account for the non-equilibrium effects that are neglected in the standard wall functions.

4.18.2.3.1. Standard Wall Functions vs. Non-Equilibrium Wall Functions

Because of the capability to partly account for the effects of pressure gradients, the non-equilibrium wall functions are recommended for use in complex flows involving separation, reattachment, and impingement where the mean flow and turbulence are subjected to pressure gradients and rapid changes. In such flows, improvements can be obtained, particularly in the prediction of wall shear (skin-friction coefficient) and heat transfer (Nusselt or Stanton number).

Important:

Non-equilibrium wall functions are available with the following turbulence closures:

- k-ε models
- Reynolds Stress Transport models

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

4.18.2.3.2. Limitations of the Wall Function Approach

The standard wall functions give reasonable predictions for the majority of high-Reynolds number wall-bounded flows. The non-equilibrium wall functions further extend the applicability of the wall function approach by including the effects of pressure gradient; however, the above wall functions become less reliable when the flow conditions depart too much from the ideal conditions underlying the wall functions. Examples are as follows:

- Pervasive low-Reynold-number or near-wall effects (for example, flow through a small gap or highly viscous, low-velocity fluid flow).
- Massive transpiration through the wall (blowing/suction).
- Severe pressure gradients leading to boundary layer separations.
- Strong body forces (for example, flow near rotating disks, buoyancy-driven flows).
- High three-dimensionality in the near-wall region (for example, Ekman spiral flow, strongly skewed 3D boundary layers).

If any of the above listed features prevail in the flow you are modeling, and if it is considered critically important for the success of your simulation, you must employ the near-wall modeling approach combined with the adequate mesh resolution in the near-wall region. For such situations, Ansys Fluent provides the enhanced wall treatment (available for the k-ε and the RSM models), as well as the Menter-Lechner near-wall treatment (available for the k-ε model).

4.18.2.4. Enhanced Wall Treatment ε-Equation (EWT-ε)

Enhanced Wall Treatment for the ε-equation is a near-wall modeling method that combines a two-layer model with so-called enhanced wall functions. If the near-wall mesh is fine enough to be able to resolve the viscous sublayer (typically with the first near-wall node placed at $y^+ \approx 1$), then the enhanced wall treatment will be identical to the traditional two-layer zonal model (see below for details). However, the restriction that the near-wall mesh must be sufficiently fine everywhere might impose too large a computational requirement. Ideally, one would like to have a near-wall formulation that can be used with coarse meshes (usually referred to as wall-function meshes) as well as fine meshes (low-Reynolds number meshes). In addition, excessive error should not be incurred for the intermediate meshes where the first near-wall node is placed neither in the fully turbulent region, where the wall functions are suitable, nor in the direct vicinity of the wall at $y^+ \approx 1$, where the low-Reynold-number approach is adequate.

To achieve the goal of having a near-wall modeling approach that will possess the accuracy of the standard two-layer approach for fine near-wall meshes and that, at the same time, will not significantly reduce accuracy for wall-function meshes, Ansys Fluent can combine the two-layer model with enhanced wall functions, as described in the following sections.

4.18.2.4.1. Two-Layer Model for Enhanced Wall Treatment

In Ansys Fluent's near-wall model, the viscosity-affected near-wall region is completely resolved all the way to the viscous sublayer. The two-layer approach is an integral part of the enhanced wall treatment and is used to specify both ε and the turbulent viscosity in the near-wall cells. In this approach, the whole domain is subdivided into a viscosity-affected region and a fully-turbulent

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

region. The demarcation of the two regions is determined by a wall-distance-based, turbulent Reynolds number, Re_y, defined as

$$Re_y \equiv \frac{\rho y \sqrt{k}}{\mu} \tag{4.359}$$

where y is the wall-normal distance calculated at the cell centers. In Ansys Fluent, y is interpreted as the distance to the nearest wall:

$$y \equiv \min_{\vec{r}_w \in \Gamma_w} \|\vec{r} - \vec{r}_w\| \tag{4.360}$$

where $\vec{r}$ is the position vector at the field point, and $\vec{r}_w$ is the position vector of the wall boundary. Γ_w is the union of all the wall boundaries involved. This interpretation allows y to be uniquely defined in flow domains of complex shape involving multiple walls. Furthermore, y defined in this way is independent of the mesh topology.

In the fully turbulent region ($Re_y > Re_y^*$; $Re_y^* = 200$), the k-ε models or the RSM (described in Standard, RNG, and Realizable k-ε Models (p. 49) and Reynolds Stress Model (RSM) (p. 96)) are employed.

In the viscosity-affected near-wall region ($Re_y < Re_y^*$), the one-equation model of Wolfstein [713] (p. 1094) is employed. In the one-equation model, the momentum equations and the k equation are retained as described in Standard, RNG, and Realizable k-ε Models (p. 49) and Reynolds Stress Model (RSM) (p. 96). However, the turbulent viscosity, μ_t, is computed from

$$\mu_{t,2layer} = \rho C_\mu \ell_\mu \sqrt{k} \tag{4.361}$$

where the length scale that appears in Equation 4.361 (p. 146) is computed from [103] (p. 1058)

$$\ell_\mu = y C_\ell^* \left(1 - e^{-Re_y / A_\mu}\right) \tag{4.362}$$

The two-layer formulation for turbulent viscosity described above is used as a part of the enhanced wall treatment, in which the two-layer definition is smoothly blended with the high-Reynolds number μ_t definition from the outer region, as proposed by Jongen [283] (p. 1069):

$$\mu_{t,enh} = \lambda_\varepsilon \mu_t + \left(1 - \lambda_\varepsilon\right) \mu_{t,2layer} \tag{4.363}$$

where μ_t is the high-Reynolds number definition as described in Standard, RNG, and Realizable k-ε Models (p. 49) or Reynolds Stress Model (RSM) (p. 96) for the k-ε models or the RSM. A blending function, λ_ε, is defined in such a way that it is equal to unity away from walls and is zero in the vicinity of the walls. The blending function has the following form:

$$\lambda_\varepsilon = \frac{1}{2}\left[1 + \tanh\left(\frac{Re_y - Re_y^*}{A}\right)\right] \tag{4.364}$$

The constant A determines the width of the blending function. By defining a width such that the value of λ_ε will be within 1% of its far-field value given a variation of ΔRe_y, the result is

$$A = \frac{\left|\Delta Re_y\right|}{artanh\left(0.98\right)} \tag{4.365}$$

Typically, ΔRe_y would be assigned a value that is between 5% and 20% of Re_y^*. The main purpose of the blending function λ_ε is to prevent solution convergence from being impeded when the

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

value of μ_t obtained in the outer layer does not match with the value of μ_t returned by the Wolfstein model at the edge of the viscosity-affected region.

The ε field in the viscosity-affected region is computed from

$$\varepsilon = \frac{k^{3/2}}{\ell_\varepsilon} \tag{4.366}$$

The length scales that appear in Equation 4.366 (p. 147) are computed from Chen and Patel [103] (p. 1058):

$$\ell_\varepsilon = y C_\ell^* \left(1 - e^{-Re_y / A_\varepsilon}\right) \tag{4.367}$$

If the whole flow domain is inside the viscosity-affected region ($Re_y < 200$), ε is not obtained by solving the transport equation; it is instead obtained algebraically from Equation 4.366 (p. 147). Ansys Fluent uses a procedure for the blending of ε that is similar to the μ_t-blending in order to ensure a smooth transition between the algebraically-specified ε in the inner region and the ε obtained from solution of the transport equation in the outer region.

The constants in Equation 4.362 (p. 146) and Equation 4.367 (p. 147), are taken from [103] (p. 1058) and are as follows:

$$C_\ell^* = \kappa C_\mu^{-3/4}, A_\mu = 70, A_\varepsilon = 2C_\ell^* \tag{4.368}$$

4.18.2.4.2. Enhanced Wall Treatment for Momentum and Energy Equations

To have a method that can extend its applicability throughout the near-wall region (that is, viscous sublayer, buffer region, and fully-turbulent outer region) it is necessary to formulate the law-of-the wall as a single wall law for the entire wall region. Ansys Fluent achieves this by blending the linear (laminar) and logarithmic (turbulent) law-of-the-wall using a function suggested by Kader [285] (p. 1069):

$$u^+ = e^{\Gamma} u_{lam}^+ + e^{1/\Gamma} u_{turb}^+ \tag{4.369}$$

where the blending function is given by:

$$\Gamma = -\frac{a\left(y^+\right)^4}{1 + by^+} \tag{4.370}$$

where $a = 0.01$ and $b = 5$.

Similarly, the general equation for the derivative $\frac{du^+}{dy^+}$ is

$$\frac{du^+}{dy^+} = e^{\Gamma} \frac{du_{lam}^+}{dy^+} + e^{1/\Gamma} \frac{du_{turb}^+}{dy^+} \tag{4.371}$$

This approach allows the fully turbulent law to be easily modified and extended to take into account other effects such as pressure gradients or variable properties. This formula also guarantees the correct asymptotic behavior for large and small values of y^+ and reasonable representation of velocity profiles in the cases where y^+ falls inside the wall buffer region ($3 < y^+ < 10$).

The enhanced wall functions were developed by smoothly blending the logarithmic layer formulation with the laminar formulation. The enhanced turbulent law-of-the-wall for compressible flow with heat transfer and pressure gradients has been derived by combining the approaches of White and Cristoph [703] (p. 1093) and Huang et al. [259] (p. 1068):

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$\frac{du^+_{turb}}{dy^+} = \frac{1}{\kappa y^+}\left[S'\left(1-\beta u^+ - \gamma\left(u^+\right)^2\right)\right]^{1/2} \tag{4.372}$$

where

$$S' = \begin{cases} 1+\alpha y^+ & \text{for } y^+ < y^+_s \\ 1+\alpha y^+_s & \text{for } y^+ \geq y^+_s \end{cases} \tag{4.373}$$

and

$$\alpha \equiv \frac{\nu_w}{\tau_w u^*}\frac{dp}{dx} = \frac{\mu}{\rho\left(u^*\right)^3}\frac{dp}{dx} \tag{4.374}$$

$$\beta \equiv \frac{\sigma_t q_w u^*}{c_p \tau_w T_w} = \frac{\sigma_t q_w}{\rho c_p u^* T_w} \tag{4.375}$$

$$\gamma \equiv \frac{\sigma_t\left(u^*\right)^2}{2c_p T_w} \tag{4.376}$$

where y^+_s is the location at which the log-law slope is fixed. By default, $y^+_s = 60$. The coefficient α in Equation 4.372 (p. 148) represents the influences of pressure gradients while the coefficients β and γ represent the thermal effects. Equation 4.372 (p. 148) is an ordinary differential equation and Ansys Fluent will provide an appropriate analytical solution. If α, β, and γ all equal 0, an analytical solution would lead to the classical turbulent logarithmic law-of-the-wall.

The laminar law-of-the-wall is determined from the following expression:

$$\frac{du^+_{lam}}{dy^+} = 1+\alpha y^+ \tag{4.377}$$

Note that the above expression only includes effects of pressure gradients through α, while the effects of variable properties due to heat transfer and compressibility on the laminar wall law are neglected. These effects are neglected because they are thought to be of minor importance when they occur close to the wall. Integration of Equation 4.377 (p. 148) results in

$$u^+_{lam} = y^+\left(1+\frac{\alpha}{2}y^+\right) \tag{4.378}$$

Enhanced thermal wall functions follow the same approach developed for the profile of u^+. The unified wall thermal formulation blends the laminar and logarithmic profiles according to the method of Kader [285] (p. 1069):

$$T^+ \equiv \frac{(T_w - T_P)\rho c_p u^*}{\dot{q}} = e^{\Gamma} T^+_{lam} + e^{1/\Gamma} T^+_{turb} \tag{4.379}$$

where the notation for T_P and $\dot{q}$ is the same as for standard thermal wall functions (see Equation 4.343 (p. 140)). Furthermore, the blending factor Γ is defined as

$$\Gamma = -\frac{a\left(Pr y^+\right)^4}{1+bPr^3 y^+} \tag{4.380}$$

where Pr is the molecular Prandtl number, and the coefficients a and b are defined as in Equation 4.370 (p. 147).

Apart from the formulation for T^+ in Equation 4.379 (p. 148), the enhanced thermal wall functions follow the same logic as for standard thermal wall functions (see Energy (p. 139)), resulting in the following definition for turbulent and laminar thermal wall functions:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$T^+_{lam} = Pr\left(u^+_{lam} + \frac{\rho u^*}{2\dot{q}}u^2\right) \tag{4.381}$$

$$T^+_{turb} = Pr_t\left\{u^+_{turb} + P + \frac{\rho u^*}{2\dot{q}}\left[u^2 + \left(\frac{Pr}{Pr_t} - 1\right)\left(u^+_c\right)^2\left(u^*\right)^2\right]\right\} \tag{4.382}$$

where the quantity u^+_c is the value of u^+ at the fictitious "crossover" between the laminar and turbulent region. The function P is defined in the same way as for the standard wall functions.

A similar procedure is also used for species wall functions when the enhanced wall treatment is used. In this case, the Prandtl numbers in Equation 4.381 (p. 149) and Equation 4.382 (p. 149) are replaced by adequate Schmidt numbers. See Species (p. 141) for details about the species wall functions.

The boundary conditions for the turbulence kinetic energy are similar to the ones used with the standard wall functions (Equation 4.349 (p. 142)). However, the production of turbulence kinetic energy, G_k, is computed using the velocity gradients that are consistent with the enhanced law-of-the-wall (Equation 4.369 (p. 147) and Equation 4.371 (p. 147)), ensuring a formulation that is valid throughout the near-wall region.

Important:

The enhanced wall treatment is available for all ε-equation models (except the Quadratic RSM).

An alternative formulation for k-ε type models is the Menter-Lechner treatment, which is a y^+-insensitive near-wall treatment. y^+-insensitive near-wall treatments are also applied by default for the following turbulence models:

- all ω-equation models
- the Spalart-Allmaras model

Note that an alternative blending compared to the Kader-blending for u^+ and $\frac{du^+_{turb}}{dy^+}$ is used for the Spalart-Allmaras model and the improved near-wall treatment for turbulence models based on the ω-equation, which is the new default near-wall treatment.

4.18.2.5. Menter-Lechner ε-Equation (ML-ε)

Historically, there are two approaches to model flow near the wall:

- wall function approach
- low-Reynolds number model

The wall function requires the cell center of the first grid point to lay in the logarithmic layer, whereas the low-Re formulation requires an integration to the wall using a resolution of $y^+ < 1$. Both approaches produce large errors if used outside of their range of validity. The wall function method deteriorates under mesh refinement, whereas the low-Re formulation results in inaccurate wall values for the wall shear stress (and heat transfer) for meshes slightly coarser than $y^+ = 1$.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

In order to provide less sensitive formulations to the CFD user, wall models have been developed that are insensitive to y^+. This means that the computed wall values (shear stress and heat transfer) are largely independent of the y^+ value provided by the mesh. Any y^+-insensitive wall treatment reverts back to its underlying low-Re formulation if the mesh is sufficiently fine, and reverts to a wall function formulation for coarse meshes. One of the main obstacles in the formulation of such wall treatments for the ε-equation is that no suitable low-Re formulation has been available. While many such models have been developed and published, each formulation suffered from one or more of the following problems:

- The formulation is complicated, involving numerous highly nonlinear damping terms.
- The formulation is not numerically robust for complex applications.
- The formulation produces multiple solutions for the same application (that is, non-unique solutions).
- The formulation produces "pseudo-transitional" results (that is, unphysical laminar zones).

Because of these problems, the model formulation of choice in today's industrial codes for the ε-equation is a so-called two-layer formulation. It avoids the solution of the ε-equation in the viscous sublayer and overwrites it with an algebraic formulation based on a simple mixing length model.

In Fluent, the two-layer model is the basis of the Enhanced Wall Treatment (EWT-ε), which is a y^+-insensitive formulation for all ε-equation based models. In the two-layer approach, the fluid domain is subdivided into a viscosity-affected region and a fully turbulent region. The blending of the two regions is determined by a turbulent Reynolds number Re_y (see Enhanced Wall Treatment ε-Equation (EWT-ε) (p. 145) for more details).

However, using a turbulent Reynolds number Re_y for the demarcation of the flow regime has some drawbacks:

- Regions with very low values of turbulence kinetic energy might easily have a turbulent Reynolds number smaller than 200. These regions will therefore be treated with a near-wall formulation, even though they might be far away from the wall (for example, regions with a very low turbulence level).
- The Wolfstein model is not consistent with the ε-equation for non-equilibrium (pressure gradient) flows. The solution of the combination therefore depends on the switching location.
- For a mesh that is coarse with a y^+ of the first cell near the switching location, the model has a tendency to oscillate, as it switches back and forth between time steps. This oscillation prevents convergence.

In order to avoid such drawbacks, the Menter-Lechner near-wall treatment has been developed as an alternative formulation that is not based on the two-layer approach. It also uses a new low-Re formulation that is designed to avoid the previously listed deficiencies of existing k-ε low-Re formulations.

The goal of a y^+-insensitive near-wall treatment is the y^+-independent prediction of the wall shear stress and wall heat flux (assuming a sufficient resolution of the boundary layer). The formulation should switch gradually from wall functions to a low-Re formulation when the mesh is refined. This also requires a blending of various quantities between the viscous sublayer and the logarithmic region.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

4.18.2.5.1. Momentum Equations

The wall shear stress τ_w is needed as a boundary condition and is calculated as:

$$\tau_w = \rho u^* u_\tau \tag{4.383}$$

where ρ denotes the density. Both of the friction velocities (u_τ and u^*) are blended between the viscous sublayer and the logarithmic region. The following formulation is used for the blending of u^*:

$$u^* = \left[\left(\mu \left| \frac{\Delta U_{\tan}}{\Delta y} \right| + C_\mu^{1/2} \rho k \right) \frac{1}{\rho} \right]^{1/2} \tag{4.384}$$

For the friction velocity u_τ, the following formulation is used:

$$u_\tau = U_{\tan} \left[\left(u_{lam}^+ \right)^{-4} + \left(u_{turb}^+ \right)^{-4} \right]^{1/4} \tag{4.385}$$

4.18.2.5.2. k-ε Turbulence Models

The main idea of the Menter-Lechner near-wall treatment is to add a source term to the transport equation of the turbulence kinetic energy k that accounts for near-wall effects. The standard k-ε model is modified as shown in the following equations (for the sake of simplicity, buoyancy effects are not included):

$$\frac{\partial}{\partial t}(\rho k) + \frac{\partial}{\partial x_i}(\rho k u_i) - \frac{\partial}{\partial x_j}\left[\left(\mu + \frac{\mu_t}{\sigma_k} \right) \frac{\partial k}{\partial x_j} \right] = G_k - \rho\varepsilon + S_{\text{near-wall}} \tag{4.386}$$

$$\frac{\partial}{\partial t}(\rho \varepsilon) + \frac{\partial}{\partial x_i}(\rho \varepsilon u_i) - \frac{\partial}{\partial x_j}\left[\left(\mu + \frac{\mu_t}{\sigma_\varepsilon} \right) \frac{\partial k}{\partial x_j} \right] = C_{1\varepsilon} \frac{\varepsilon}{k} G_k - C_{2\varepsilon} \rho \frac{\varepsilon^2}{k} \tag{4.387}$$

$$\mu_t = \rho C_\mu \frac{k^2}{\varepsilon} \tag{4.388}$$

where $C_{1\varepsilon}$=1.44, $C_{2\varepsilon}$=1.92, C_μ=0.09, σ_k=1.0, and σ_ε=1.3.

The additional source term $S_{\text{near-wall}}$ is active only in the viscous sublayer and accounts for low-Reynolds number effects. It automatically becomes zero in the logarithmic region. The exact formulation of this source term is at this point proprietary and is therefore not provided here.

4.18.2.5.3. Iteration Improvements

In combination with the Menter-Lechner near-wall treatment, the iterative treatment and linearization of the k-ε two-equation model has been improved. This modification is activated by default when this near-wall treatment is used.

The Menter-Lechner near-wall treatment can be used together with the standard, realizable, and RNG k-ε turbulence models.

4.18.2.6. User-Defined Wall Functions

This option is only available when one of the k-ε model is enabled. Selecting the **User-Defined Wall Functions** under **Near-wall Treatment** allows you to hook a **Law-of-the-Wall** UDF. For more

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

information about user-defined wall functions, see `DEFINE_WALL_FUNCTIONS` in the Fluent Customization Manual.

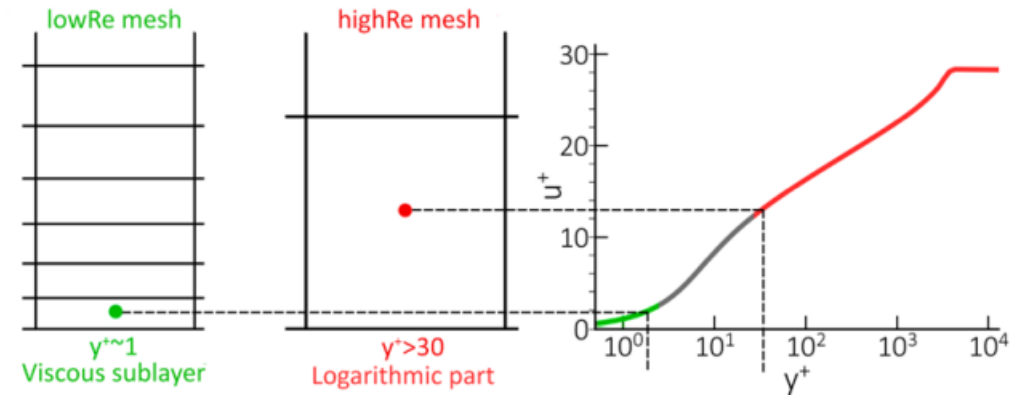
Important:

User-defined wall functions are available with the k-ε turbulence closure model:

4.18.3. y+-Insensitive Near-Wall Treatment for ω-based Turbulence Models

An essential point in the implementation of turbulence models is the near-wall treatment. There are two distinct branches for setting the wall boundary conditions depending on first near wall grid step (see Figure 4.17: Computational meshes for low and high Reynolds number cases (p. 152)). The first constitutes a low Reynolds number formulation (lowRe) where the first near wall grid step is inside the viscous sublayer of the boundary layer ($y^+ < 5$). The second corresponds to high Reynolds number (highRe) meshes where wall functions are activated. In this case, the first near wall grid step corresponds to the logarithmic profile ($y^+ > 30$). Between these two cases, the so-called buffer layer is located. Ideally simulations should be computed with meshes where y^+ is 1 , however this requirement is not always easy to achieve, especially in industrial computations at high Reynolds numbers and with complex geometries. It is also not always possible to create pure wall function meshes with $y^+ > 30$ on the entire geometry. Wall-formulations have therefore been developed which smoothly blend between the two limits. Such formulations are termed "y^+-insensitive near-wall treatment".

Figure 4.17: Computational meshes for low and high Reynolds number cases



As the formulation suggests, these methods are not entirely "y^+-insensitive" and a small difference in wall shear stress values is typically observed in the buffer layer, compared to the reference fine grid solution.

The goal of the y^+-insensitive near-wall treatment is to minimize these variations and to develop wall boundary conditions which allow for solutions to be as insensitive to near-wall mesh resolution as possible for the wall shear stress and wall heat flux assuming a sufficient overall resolution of the boundary layer.

In turbulence models based on the ω-equation, the wall shears stress is controlled by the values specified for ω at the wall. In Ansys Fluent, two y^+-insensitive near-wall treatments are available:

- `correlation`: blends between analytical expressions of the viscous sublayer and logarithmic region depending on y^+ (default).

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

- `tabulated`: employs a table look-up for the wall value of ω to minimize analytical interpolation errors between the two regimes.

Both formulations were calibrated using fully developed plane Couette flow and will be outlined in the following sections. Beside the treatment of wall ω other quantities also require a gradual shift from wall functions to a low-Re formulation when the mesh is refined.

4.18.3.1. Momentum Equations

The wall shear stress τ_w is needed as a boundary condition and calculated in the following way:

$$\tau_w = \rho u_\tau u^* \tag{4.389}$$

where ρ denotes the density. Equation 4.389 (p. 153) is based on the friction velocity u_τ and the alternative velocity scale u^*, which has been introduced to improve the behavior at separation. Both quantities are blended between the viscous sublayer and the logarithmic region.

The friction velocity u_τ is given by:

$$u_\tau = \frac{U_{\tan}}{\left[\left(u^+_{lam}\right)^{-4} + \left(u^+_{turb}\right)^{-4}\right]^{-1/4}} \tag{4.390}$$

where

$$u^+_{lam} = y^+ \tag{4.391}$$

$$u^+_{turb} = \frac{1}{\kappa}\ln\left[E \cdot MAX\left(y^+, 0.2\right)\right] \tag{4.392}$$

with κ=0.4187 and E=9.793. $U_{\tan}$ represents the mean velocity tangent to the wall at the wall-adjacent cell centroid. The non-dimensional distance from the wall y^+ is defined using u^*:

$$y^+ = \Delta y \frac{\rho u^*}{\mu} \tag{4.393}$$

Δy denotes the distance from the wall at the wall-adjacent cell centroid and μ the molecular viscosity.

For the blending of the alternative velocity scale u^* the following formulation is used:

$$u^* = \left[\left(\mu \left|\frac{\Delta U_{\tan}}{\Delta y}\right| + \left(C_\mu^{1/2}\rho k\right)\right)\frac{1}{\rho}\right]^{1/2} \tag{4.394}$$

where k denotes the turbulence kinetic energy at the wall-adjacent cell centroid.

4.18.3.2. Turbulence

For the transport equation of the turbulence kinetic energy, a zero flux boundary condition is specified not only in the logarithmic region, but also (artificially) for the integration down to the wall. The wall production term is blended between the viscous sublayer and logarithmic region. The production term can be written using u^+ and y^+ in the following way:

$$P_k = \mu_t \left(\frac{\tau_w}{\mu}\frac{\partial u^+}{\partial y^+}\right)^2 \tag{4.395}$$

which gives the following relations for the viscous sublayer and logarithmic region:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$P_{k,lam}=\frac{\rho k}{\omega}\left(\frac{\tau_w}{\mu}\right)^2 \tag{4.396}$$

$$P_{k,turb}=\frac{\tau_w^2}{\mu}\frac{\partial u_{turb}^+}{\partial y^+} \tag{4.397}$$

Note that $\partial u_{turb}^+ / \partial y^+$ is not based on a gradient computation, but directly derived from the log-law relation in Equation 4.392 (p. 153).

The blended form of the wall production term finally reads:

$$P_{k,wall}=\left(P_{k,lam}^{-1}+P_{k,turb}^{-1}\right)^{-1} \tag{4.398}$$

In the transport equation of the specific dissipation rate, the value of ω at the wall is prescribed. The wall value is blended between the relations of the viscous sublayer and logarithmic region in the following way:

$$\omega_{\text{wall}}=\frac{\rho\left(u^*\right)^2}{\mu}\omega_{\text{wall}}^+ \tag{4.399}$$

where ρ and μ denote the density and the dynamic viscosity, respectively. u* represents the alternative velocity scale given in Equation 4.394 (p. 153).

Analytical solutions for ω_{wall}^+ can be given for both the laminar sublayer and the logarithmic region. This strategy can used to define a y^+-insensitive near wall treatment by blending between these two analytical expressions depending on y^+:

$$\omega_{wall}^+=\omega_{lam}^+\cdot\left[1+\left(\frac{\omega_{turb}^+}{\omega_{lam}^+}\right)^{C_{\text{exp}}}\right]^{1/C_{\text{exp}}} \tag{4.400}$$

$$\omega_{lam}^+=\frac{C_{calib}\cdot 6}{\beta_i\left(y^+\right)^2} \tag{4.401}$$

$$\omega_{turb}^+=\frac{1}{\sqrt{\beta_\infty^*}}\cdot\frac{du_{turb}^+}{dy^+} \tag{4.402}$$

This formulation is the current default treatment in Ansys Fluent and is labelled further-on as "`correlation`". Incompressible plane Couette flow at Reynolds number 10^6 has been used to calibrate the entire formulation for momentum and turbulence equations in order to minimize the variation of the wall shear stress over y^+. This leads to a value of 1/3 for C_{calib} and 1.3 for C_{exp}. However, due to its analytical expression, this formulation offers only limited possibilities for fine tuning.

A further improvement was achieved by using a table look-up for ω_{wall}^+ for a wide range of y^+ values. Incompressible plane Couette flow has again been used to calibrate this formulation in order to minimize the variation of the wall shear stress over y^+. For this purpose, a fine grid solution was created, and the resulting wall shear stress was used as a reference on coarser grids. A table was created by running the plane Couette flow for various meshes, storing the optimal ω_{wall} and then transforming the obtained data sets into an efficient table look-up for ω_{wall}^+.

Specific tables have been created for BSL, SST, and GEKO turbulence models and this method is further-on labelled as "`tabulated`". All other ω-based turbulence models are using the SST table

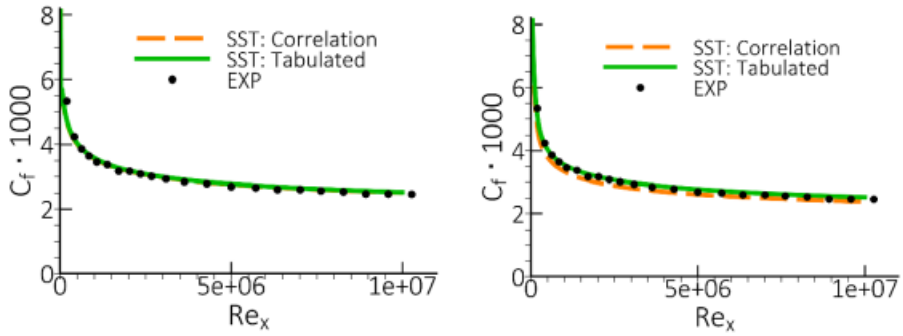
Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

look-up so that for example, the Transition SST, DES-SST, and SAS-SST (all implemented as separate models) are using a consistent table.

Both formulations agree well for high quality meshes with small values of y+ (y+ < 1) and moderate expansion ratios. Although these meshes are generally recommended, it might not always be possible to create such meshes due to the complexity of the applications. The table look-up offers a more accurate prediction of the wall shear stress, especially for poorer meshes having larger values of y+ and/or higher expansion ratios.

As an example of this improvement, the results of a zero pressure gradient flat plate boundary layer flow are presented below. The setup is based on the experiment of Wieghardt & Tillmann [780] (p. 1098) and the Reynolds number based on inlet velocity and length of the flat plate is $Re=10^7$. Figure 4.18: Comparison of the skin friction distribution on a fine (y+ ~ 0.1, left) and coarse (y+ ~ 22, right) mesh for the 'correlation' and 'tabulated' near wall treatments with SST. (p. 155) shows the skin friction distribution for a fine (y+ ~ 0.1) and coarse (y+ ~ 22) mesh computed with the SST turbulence model. The visual difference between the 2 formulations appears small and confirms that the correlation model is already reasonably close.

Figure 4.18: Comparison of the skin friction distribution on a fine (y+ ~ 0.1, left) and coarse (y+ ~ 22, right) mesh for the 'correlation' and 'tabulated' near wall treatments with SST.



However, looking at the overall drag reveals that the "tabulated" (new) model is superior for this case. The error in drag prediction is defined as

$$c_{d,error} = \frac{c_d - c_{d,ref}}{c_{d,ref}} \tag{4.403}$$

where the reference solution $c_{d,ref}$ corresponds to the solution on a very fine mesh with y+ ~ 0.001. The results are summarized in Table 1 and it can be seen that the error is strongly reduced on the coarse mesh for the "tabulated" near wall treatment:

Table 4.3: Error in drag prediction on a fine (y+ ~ 0.1) and coarse mesh (y+ ~ 22) for the 'correlation' and 'tabulated' near wall treatments with SST, relative to (y+ ~ 0.001) mesh

Mesh	$c_{d,ref}$	
	SST: correlation	SST: tabulated
$y^+ \sim 0.1$	0.47%	0.05%
$y^+ \sim 22$	5.86%	0.02%

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

4.18.3.3. Energy

The near wall treatment of the energy equation follows a Kader blending method between the viscous sublayer and logarithmic law of the wall as outlined by Equation 4.379 (p. 148).

4.18.3.4. Limitations

- For the `tabulated` option, there are specially calibrated tables for BSL, SST, and GEKO. All other omega-based turbulence models are using the SST table look-up so that for example, the Transition SST, DES-SST and SAS-SST (all implemented as separate models) are using a consistent table look-up.
- Both the formulations can also be combined with the Standard Roughness Model.

Note:

Note: The High Roughness (Icing) Model only works on y+ ~ 1 meshes and uses a separate near wall treatment for the walls where it is specified.

4.18.4. LES Near-Wall Treatment

For the LES models, following wall function are available:

4.18.4.1. Kader Blending Wall Functions

4.18.4.2. Harmonic Blending Wall Functions Based on r+

4.18.4.3. Werner and Wengle Wall Treatment

4.18.4.1. Kader Blending Wall Functions

Thie widely used blending of the laminar sublayer and logarithmic laws of Kader [285] (p. 1069) is the default option in Fluent.

When the mesh is fine enough to resolve the laminar sublayer, the wall shear stress is obtained from the laminar stress-strain relationship:

$$\frac{\overline{u}}{u_\tau} = \frac{\rho u_\tau y}{\mu} \tag{4.404}$$

If the mesh is too coarse to resolve the laminar sublayer, it is assumed that the centroid of the wall-adjacent cell falls within the logarithmic region of the boundary layer, and the law-of-the-wall is employed:

$$\frac{\overline{u}}{u_\tau} = \frac{1}{\kappa} \ln E\left(\frac{\rho u_\tau y}{\mu}\right) \tag{4.405}$$

where κ is the von Kármán constant and $E = 9.793$. If the mesh is such that the first near-wall point is within the buffer region, then two above laws are blended in accordance with Equation 4.369 (p. 147).

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

4.18.4.2. Harmonic Blending Wall Functions Based on r+

Classical law-of-the-wall formulations are of the form:

$$u^+ = f\left(y^+\right) \tag{4.406}$$

This equation relates the unknown shear velocity u_τ with the known quantities of u, y, and ν at the near wall cell center. To compute u_τ, iterations are required and the explicit formulation is based on an alternative variable which does not contain u_τ:

$$r^+ = \frac{uy}{\nu} \tag{4.407}$$

viscous sub-layer formulation:

$$u^+_{sub} = \sqrt{r^+} \tag{4.408}$$

log-layer formulation:

$$u^+_{\log} = C_1 \ln\left(r^+ + C_2\right) \tag{4.409}$$

with C_1=2.21, C_2=4.

To blend the viscous sub-layer with the logarithmic layer, the harmonic power law is employed:

$$u^+\left(r^+\right) = \left[\left(u^+_{sub}\right)^{-\alpha} + \left(u^+_{\log}\right)^{-\alpha}\right]^{-\frac{1}{\alpha}} \tag{4.410}$$

with an exponent of α=6.

Harmonic blending wall functions can be enabled from The Viscous Model Panel (as described in Setting Up the Large Eddy Simulation Model in the *Fluent User's Guide*) or by using the following text command:

```
define/models/viscous/near-wall-treatment/harmonic-blend-rplus-wf?
```

Unlike other Wall Treatment options available for LES, r^+-based wall functions support including wall roughness effect (see Wall Roughness Effects in Turbulent Wall-Bounded Flows (p. 158)).

Note:

To maintain the full effect of the roughness on the flow when using Harmonic blending wall functions based on r^+ in combination with Rough Walls, make sure that the mesh satisfies the requirements $y_1 > k_s$.

4.18.4.3. Werner and Wengle Wall Treatment

For the LES simulations in Ansys Fluent, there is an alternative near-wall approach based on the work of Werner and Wengle [699] (p. 1093), who proposed an analytical integration of the power-law near-wall velocity distribution resulting in the following expressions for the wall shear stress:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$|\tau_w| = \begin{cases} \frac{2\mu|u_p|}{\Delta z} & \text{for } |u_p| \le \frac{\mu}{2\rho\Delta z} A^{\frac{2}{1-B}} \\ \rho\left[\frac{1-B}{2} A^{\frac{1+B}{1-B}} \left(\frac{\mu}{\rho\Delta z}\right)^{1+B} + \frac{1+B}{A}\left(\frac{\mu}{\rho\Delta z}\right)^{B} |u_p|\right]^{\frac{2}{1+B}} & \text{for } |u_p| > \frac{\mu}{2\rho\Delta z} A^{\frac{2}{1-B}} \end{cases} \tag{4.411}$$

where u_p is the wall-parallel velocity, $A = 8.3, B = 1/7$ are the constants, and Δz is the near-wall control volume length scale.

The Werner-Wengle wall functions can be enabled using the `define/models/viscous/near-wall-treatment/werner-wengle-wall-fn?` text command.

4.18.5. Wall Roughness Effects in Turbulent Wall-Bounded Flows

Experiments in roughened pipes and channels indicate that the mean velocity distribution near rough walls, when plotted in the usual semi-logarithmic scale, has the same slope ($1/\kappa$) but a different intercept (additive constant B in the log-law). Therefore, the law-of-the-wall for mean velocity modified for roughness has the form

$$\frac{u_p u^*}{\tau_w/\rho} = \frac{1}{\kappa} \ln\left(E \frac{\rho u^* y_p}{\mu}\right) - \Delta B \tag{4.412}$$

where $u^* = C_\mu^{1/4} k^{1/2}$ and

$$\Delta B = \frac{1}{\kappa} \ln f_r \tag{4.413}$$

where f_r is a roughness function that quantifies the shift of the intercept due to roughness effects.

For r^+-based wall functions for LES (see Harmonic Blending Wall Functions Based on r+ (p. 157)), the logarithmic part of the profile is shifted at rough walls using following formulation:

$$u^+_{\log} = C_1 \ln\left[r^+ + \frac{\Delta B}{u^+_{\log}} r^+ + C_2\right] - \Delta B \tag{4.414}$$

ΔB depends, in general, on the type (uniform sand, rivets, threads, ribs, mesh-wire, and so on) and size of the roughness. There is no universal roughness function valid for all types of roughness. For a sand-grain roughness and similar types of uniform roughness elements, however, ΔB has been found to be well-correlated with the nondimensional roughness height, $K_s^+ = \rho K_s u^*/\mu$, where K_s is the physical roughness height and $u^* = C_\mu^{1/4} k^{1/2}$. Analyses of experimental data show that the roughness function is not a single function of K_s^+, but takes different forms depending on the K_s^+ value. It has been observed that there are three distinct regimes:

- hydrodynamically smooth ($K_s^+ \le 2.25$)
- transitional ($2.25 < K_s^+ \le 90$)
- fully rough ($K_s^+ > 90$)

According to the data, roughness effects are negligible in the hydrodynamically smooth regime, but become increasingly important in the transitional regime, and take full effect in the fully rough regime.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

In Ansys Fluent, the whole roughness regime is subdivided into the three regimes, and the formulas proposed by Cebeci and Bradshaw based on Nikuradse's data [98] (p. 1058) are adopted to compute ΔB for each regime.

For the hydrodynamically smooth regime ($K_s^+ \leq 2.25$):

$$\Delta B = 0 \tag{4.415}$$

For the transitional regime ($2.25 < K_s^+ \leq 90$):

$$\Delta B = \frac{1}{\kappa} \ln \left[\frac{K_s^+ - 2.25}{87.75} + C_s K_s^+ \right] \times \sin \left\{ 0.4258 \left(\ln K_s^+ - 0.811 \right) \right\} \tag{4.416}$$

where C_s is a roughness constant, and depends on the type of the roughness.

In the fully rough regime ($K_s^+ > 90$):

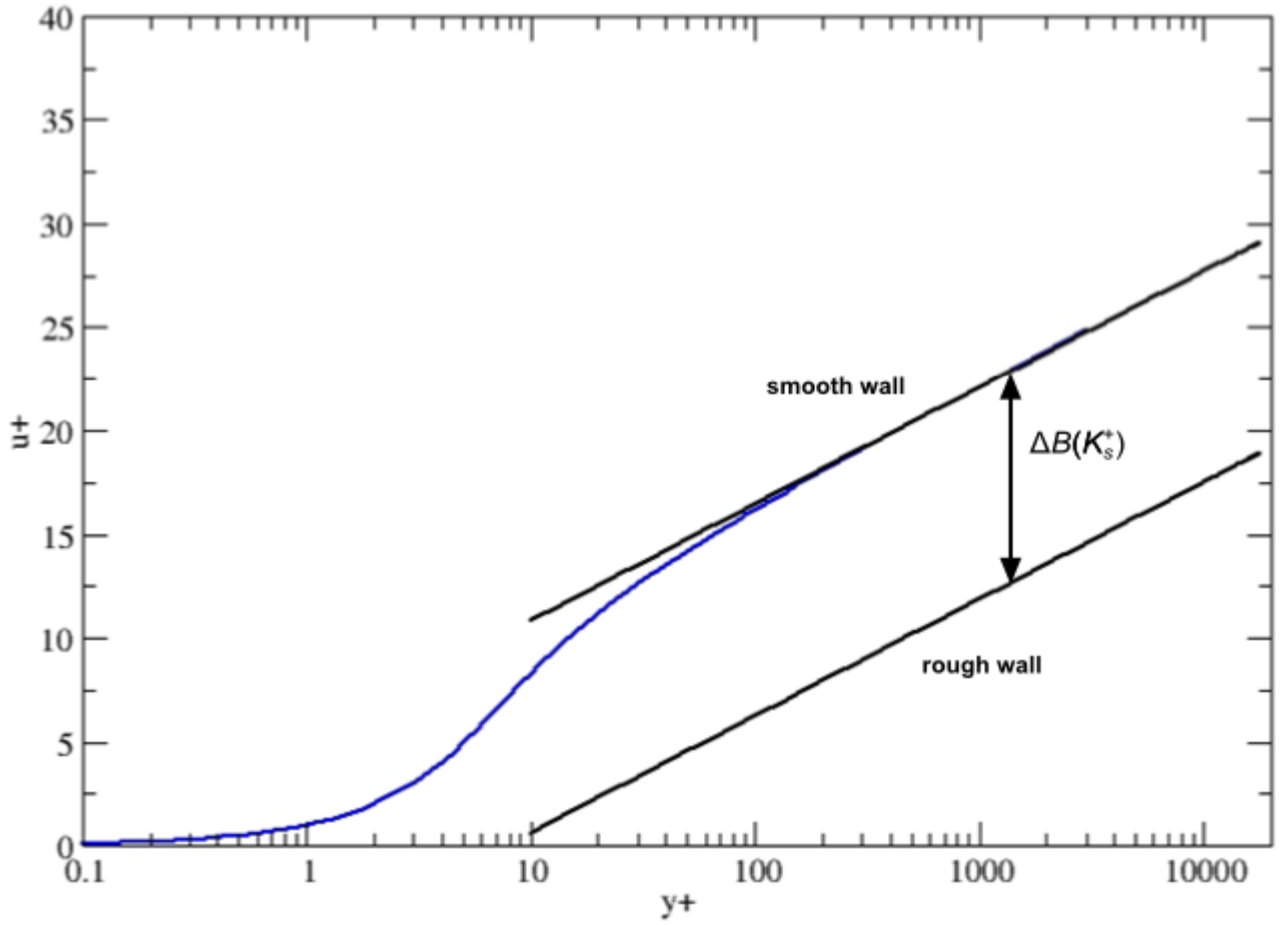
$$\Delta B = \frac{1}{\kappa} \ln \left(1 + C_s K_s^+ \right) \tag{4.417}$$

In the solver, given the roughness parameters, $\Delta B\left(K_s^+\right)$ is evaluated using the corresponding formula (Equation 4.415 (p. 159), Equation 4.416 (p. 159), or Equation 4.417 (p. 159)). The modified law-of-the-wall in Equation 4.412 (p. 158) is then used to evaluate the shear stress at the wall and other wall functions for the mean temperature and turbulent quantities.

$\Delta B\left(K_s^+\right)$ represents a downward shift of the logarithmic velocity profile, as shown in the following figure:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Figure 4.19: Downward Shift of the Logarithmic Velocity Profile



This downward shift leads to a singularity for large roughness heights and low values of y^+. Depending on the turbulence model and near wall treatment, two different approaches are used in Ansys Fluent in order to avoid this issue:

- reducing the roughness height as y^+ decreases

  The first approach consists in redefining the roughness height based on the mesh refinement:

  $$K_s^+ = \min\left(K_s^+, y^+\right) \tag{4.418}$$

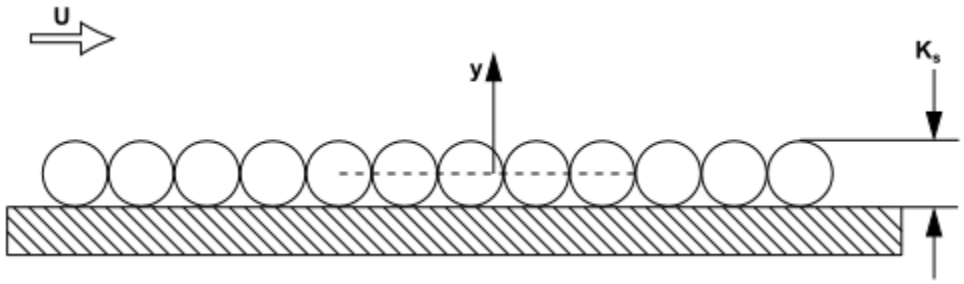
  This ensures that as y^+ approaches zero, so too does K_s^+. Therefore, the mesh requirement for rough walls in this case is $y^+ > K_s^+$, in order to maintain the full effect of the roughness on the flow.

- virtually shifting the wall

  The second approach is based on the observation that the viscous sublayer is fully established only near hydraulically smooth walls. In the transitional roughness regime, the roughness elements are slightly thicker than the viscous sublayer and start to disturb it, so that in fully rough flows, the sublayer is destroyed and viscous effects become negligible. The following figure illustrates the equivalent sand-grain roughness using a wall with a layer of closely packed spheres, which have an average roughness height representing a technical roughness with peaks and valleys of different shapes and sizes (see Schlichting and Gersten [575] (p. 1086)):

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Figure 4.20: Illustration of Equivalent Sand-Grain Roughness



It can be assumed that the roughness has a blockage effect, which is about 50% of its height (note that the figure above shows a two-dimensional cut of a three-dimensional arrangement).

It is therefore sensible to virtually shift the wall to 50% of the height of the roughness elements. This results in a corrected y^+ value for the first cell center:

$$y^+ = y^+ + K_s^+ / 2 \tag{4.419}$$

which gives about the correct displacement caused by the surface roughness. Thus the singularity issue is avoided and fine meshes can be handled correctly.

The second approach (that is, virtually shifting the wall) is the default treatment for rough walls for all two-equation turbulence models based on the ω -equation and for the following turbulence models based on the ε -equation, when they are used with standard and scalable wall functions (note that the use of scalable wall functions is recommended over the use of standard wall functions):

- standard, RNG, and realizable k - ε model
- Reynolds stress models

All other model combinations with rough walls (for example, the Spalart-Allmaras model and LES in combination with Harmonic Blending Wall Functions Based on r+ (p. 157)) have no special calibration on fine meshes, and therefore the first approach (reducing the roughness height as y^+ decreases) is used.

Note:

The advantage of the rough wall formulation using a virtual shift of the wall (Equation 4.419 (p. 161)) compared to reducing the roughness height as y^+ decreases (Equation 4.418 (p. 160)) is that it eliminates all restrictions with respect to mesh resolution near the wall, and can therefore be used on arbitrarily fine meshes.

Important:

Prior to Ansys Fluent 14, the shift described by Equation 4.419 (p. 161) was not applied when using turbulence models based on the ε -equation. You can recover the previous code behavior by using the following scheme command:

```
(rpsetvar 'ke-rough-wall-treatment-r14? #f)
(models-changed)
```

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

4.19. Curvature Correction for the Spalart-Allmaras and Two-Equation Models

One drawback of the eddy-viscosity models is that these models are insensitive to streamline curvature and system rotation, which play a significant role in many turbulent flows of practical interest. A modification to the turbulence production term is available to sensitize the following standard eddy-viscosity models to the effects of streamline curvature and system rotation:

- Spalart-Allmaras one-equation model
- Standard, RNG, and Realizable (k-ε)-models
- Standard (k-ω), BSL, SST, and Transition SST
- Scale-Adaptive Simulation (SAS), Detached Eddy Simulation with BSL / SST (DES-BSL / SST), Shielded Detached Eddy Simulation (SDES), and Stress-Blended Eddy Simulation (SBES).

Note that both the RNG and Realizable (k-ε) turbulence models already have their own terms to include rotational or swirl effects (see RNG Swirl Modification (p. 52) and Modeling the Turbulent Viscosity (p. 56), respectively, for more information). The curvature correction option should therefore be used with caution for these two models and is offered mainly for completeness for RNG and Realizable (k-ε).

Spalart and Shur [620] (p. 1089) and Shur et al. [592] (p. 1087) have derived a modification of the production term for the Spalart-Allmaras one-equation turbulence model to take streamline curvature and system rotation effects into account. Based on this work, a modification of the production term has been derived that allows you to sensitize the standard two-equation turbulence models to these effects (Smirnov and Menter [601] (p. 1088)). The empirical function suggested by Spalart and Shur [620] (p. 1089) to account for streamline curvature and system rotation effects is defined by

$$f_{rotation} = \left(1+c_{r1}\right)\frac{2r^*}{1+r^*}\left[1-c_{r3}\tan^{-1}\left(c_{r2}\tilde{r}\right)\right]-c_{r1} \tag{4.420}$$

It is used as a multiplier of the production term and has been limited in Ansys Fluent in the following way:

$$P_k \rightarrow P_k \cdot f_r \tag{4.421}$$

with

$$f_r = \max\left\{0, 1+C_{curv}\left(\tilde{f}_r-1\right)\right\} \tag{4.422}$$

$$\tilde{f}_r = \max\left\{\min\left(f_{rotation}, 1.25\right), 0\right\} \tag{4.423}$$

The original function is limited in the range from 0 corresponding to, for example, a strong convex curvature (stabilized flow, no turbulence production) up to 1.25 (strong concave curvature, enhanced turbulence production). The lower limit is introduced for numerical stability reasons, whereas the upper limit is needed to avoid over-generation of the eddy viscosity in flows with a destabilizing curvature/rotation. The specific limiter 1.25 provides a good compromise for different test cases that have been considered with the SST model (such as U-turn flow, flow in hydro cyclone, NACA 0012 wing tip vortex [601] (p. 1088)).

C_{curv} in Equation 4.422 (p. 162) has been introduced to allow you to influence the strength of the curvature correction if needed for a specific flow. By default, the coefficient is constant and the value of C_{curv} is equal to 1. It is accessible in the Viscous Model dialog box or via text command, and you may also specify an expression or a User Defined Function (see `DEFINE_CURVATURE_CORREC-`

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

TION_CCURV in the *Fluent Customization Manual*). This offers additional flexibility to provide zonal dependent values. The coefficient C_{curv} should be positive.

Assuming that all the variables and their derivatives are defined with respect to the reference frame of the calculation, which is rotating with a rate Ω^{rot}, the arguments r^* and $\tilde{r}$ of the function $f_{rotation}$ are defined in the following way:

$$r^*=\frac{S}{\Omega} \tag{4.424}$$

$$\tilde{r}=2\Omega_{ik}S_{jk}\left[\frac{DS_{ij}}{Dt}+\left(\varepsilon_{imn}S_{jn}+\varepsilon_{jmn}S_{in}\right)\Omega_m^{Rot}\right]\frac{1}{\tilde{D}} \tag{4.425}$$

Where the first term in brackets is equivalent to the second velocity gradient (in this case the Lagrangian derivative of the strain rate tensor) and the second term in the brackets is a measure of the system rotation. The strain rate and vorticity tensor are defined as

$$S_{ij}=\frac{1}{2}\left(\frac{\partial u_i}{\partial x_j}+\frac{\partial u_j}{\partial x_i}\right) \tag{4.426}$$

$$\Omega_{ij}=\frac{1}{2}\left(\frac{\partial u_i}{\partial x_j}-\frac{\partial u_j}{\partial x_i}\right)+2\varepsilon_{mji}\Omega_m^{rot} \tag{4.427}$$

with

$$S^2=2S_{ij}S_{ij} \tag{4.428}$$

$$\Omega^2=2\Omega_{ij}\Omega_{ij} \tag{4.429}$$

The denominator in Equation 4.425 (p. 163) takes the following form for the Spalart-Allmaras turbulence model (Shur et al. [592] (p. 1087)):

$$\tilde{D}=D^4 \tag{4.430}$$

$$D^2=0.5\left(S^2-\Omega^2\right) \tag{4.431}$$

and for all other turbulence models (Smirnov and Menter [601] (p. 1088)) :

$$\tilde{D}=\Omega D^3 \tag{4.432}$$

$$D^2=\max\left(S^2,0.09\omega^2\right) \tag{4.433}$$

$\frac{DS_{ij}}{Dt}$ are the components of the Lagrangian derivative of the strain rate tensor. The Einstein summation convention is used.

The empirical constants c_{r1}, c_{r2} and c_{r3} involved in Equation 4.420 (p. 162) are set equal to

- 1.0, 12.0, and 1.0, respectively, for the Spalart-Allmaras model [592] (p. 1087)
- 1.0, 2.0, and 1.0, respectively, for all other turbulence models based on the performed tests in [601] (p. 1088)

Note:

The curvature correction function strongly depends on the change in orientation of the strain rate tensor. It is therefore only meaningful in shear flow regions (curved boundary layers, mixing layers, vortex flows, etc.). In regions of low or zero shear strain rate it can show strong

variations. However, this has no consequences, as the curvature correction function is multiplied by the production term of the turbulence kinetic energy equations - which is small/zero for vanishing shear strain rate

To enable curvature correction, see Including the Curvature Correction for the Spalart-Allmaras and Two-Equation Turbulence Models in the User's Guide.

4.20. Corner Flow Correction

Turbulent flows through rectangular channels, pipes of non-circular cross-section, and wing-body-junction type geometries exhibit secondary flows in the plane normal to the main flow direction into the corner along the bisector. These flows are known as "secondary flow of the second kind" [530] (p. 1084) and are of much lower magnitude than the flow in the main direction. These secondary flows are driven by the anisotropy (meaning differences in normal stresses) of the turbulence stress tensor. They can have a subtle effect on flow separation under adverse pressure gradient conditions, where the additional momentum transport into the corner can help the flow remain attached.

Eddy-viscosity models cannot account for this effect, as they do not model the anisotropy of the normal stresses. They therefore have a tendency to over-predict corner flow separation zones. Differential Reynolds stress models are the most consistent RANS-based formulations able to take anisotropy into account, but are often too time consuming and not robust enough in complex flow situations. A simplified quadratic non-linear algebraic extension for the Spalart-Allmaras and two-equation models has been implemented that drives momentum into the corner and can reduce corner separation effects. The additional terms do not affect the basic model calibration for simple generic flows (like flat plates). A tunable model parameter is introduced which allows you to optimize the strength of the quadratic term for your application.

The corner flow correction is available for the following turbulence models:

- Spalart-Allmaras (SA,SA-DES)
- Standard k-ω model (and combination of this model with SAS)
- BSL, SST, GEKO, and Transition-SST models (and combination of these models with SAS, DES, and SBES)

For the definition of the Reynolds stress tensor in the momentum equation, a quadratic model is used based on [616] (p. 1088), but augmented for the Spalart-Allmaras model:

$$\overline{u_i'u_j'} = -2\nu_t S_{ij} - C_{corner}\frac{1.2\nu_t}{MAX\left(10^{-10}, \sqrt{(S^2+\Omega^2)/2}\right)}\left(S_{ik}\Omega_{kj} - \Omega_{ik}S_{kj}\right) \tag{4.434}$$

and for two-equation models:

$$\overline{u_i'u_j'} = \frac{2}{3}k\delta_{ij} - 2\nu_t S_{ij} - C_{corner}\frac{1.2\nu_t}{MAX\left(0.3\omega, \sqrt{(S^2+\Omega^2)/2}\right)}\left(S_{ik}\Omega_{kj} - \Omega_{ik}S_{kj}\right) \tag{4.435}$$

where ν_t denotes the kinematic eddy viscosity. The strain-rate and vorticity tensor are defined as:

$$S_{ij} = \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} - \frac{2}{3}\frac{\partial u_k}{\partial x_k}\delta_{ij}\right)$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$\Omega_{ij}=\frac{1}{2}\left(\frac{\partial u_i}{\partial x_j}-\frac{\partial u_j}{\partial x_i}\right).$$

with $S=\sqrt{2S_{ij}S_{ij}}$ and $\Omega=\sqrt{2\Omega_{ij}\Omega_{ij}}$.

The factor 1.2 is from 0.3*4, where 0.3 is the model constant C_{cr1} introduced by Spalart [616] (p. 1088) and 4 is due to the utilization of strain-rate and vorticity tensors in Equation 4.434 (p. 164) and Equation 4.435 (p. 164). Instead of giving access to C_{cr1}, a tunable parameter C_{corner} is available in Fluent.

By default, the coefficient C_{corner} is constant and the value is equal to 1. It is accessible in the Viscous Model dialog box or via text command. There is also the possibility to specify an expression or a User Defined Function which offers additional flexibility to provide zonal-dependent values.

Note:

One-equation models like Spalart-Allmaras are missing the $\frac{2}{3}k\delta_{ij}$- term which can lead to negative normal stresses.

To enable corner flow correction, see Including Corner Flow Correction in the *Fluent User's Guide*.

4.21. Production Limiters for Two-Equation Models

A disadvantage of standard two-equation turbulence models is the excessive generation of the turbulence energy, G_k, in the vicinity of stagnation points. In order to avoid the buildup of turbulent kinetic energy in the stagnation regions, the production term in the turbulence equations can be limited by one of the two formulations presented below.

The first formulation takes the following form (Menter [425] (p. 1077)):

$$G_k=\min\left(G_k, C_{\lim}\rho\varepsilon\right) \tag{4.436}$$

where the coefficient $C_{\lim}$ has a default value of 10. This limiter does not affect the shear layer performance of the model, but avoids the stagnation point buildup in aerodynamic simulations. The first formulation is available for the following turbulence models:

- Standard, RNG, and Realizable (k-ε)-models
- Standard (k-ω), BSL, SST, and Transition SST
- Scale-Adaptive Simulation (SAS)
- Detached Eddy Simulation with BSL / SST (DES-BSL / SST) and with Realizable (k-ε)-models (DES-rke)
- Shielded Detached Eddy Simulation (SDES)
- Stress-Blended Eddy Simulation (SBES)

By default, this limiter is enabled for all turbulence models based on the ω equation.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Using the standard Boussinesq approximation for the Reynolds stress tensor, the production term G_k for an incompressible flow can be expressed as:

$$G_k=\mu_t S^2 \tag{4.437}$$

where S is the magnitude of the strain rate:

$$S=\sqrt{2S_{ij}S_{ij}} \tag{4.438}$$

The strain rate tensor S_{ij} is calculated as:

$$S_{ij}=\frac{1}{2}\left(\frac{\partial U_i}{\partial x_j}+\frac{\partial U_j}{\partial x_i}\right) \tag{4.439}$$

The second formulation is based on the work of Kato and Launder ([292] (p. 1070)) who noticed that the excessive level of turbulence kinetic energy production is caused by the very high level of shear strain rate S in the stagnation regions. As the flow field near a stagnation point is nearly irrotational, with a very small vorticity rate Ω, the alternative form of the production term proposed by Kato-Launder is written as:

$$G_k=\mu_t S\Omega \tag{4.440}$$

where Ω is the magnitude of the vorticity rate:

$$\Omega=\sqrt{2\Omega_{ij}\Omega_{ij}} \tag{4.441}$$

where the vorticity tensor Ω_{ij} is calculated as:

$$\Omega_{ij}=\frac{1}{2}\left(\frac{\partial U_i}{\partial x_j}-\frac{\partial U_j}{\partial x_i}\right) \tag{4.442}$$

In a simple shear flow, S and Ω are equal and thus Equation 4.437 (p. 166) is recovered. The Kato-Launder formulation for the production term is available for the following turbulence models:

- Standard, and RNG (k-ε)-models
- Standard (k-ω), BSL, SST, and Transition SST
- Scale-Adaptive Simulation (SAS)
- Detached Eddy Simulation with BSL / SST (DES-BSL / SST)
- Shielded Detached Eddy Simulation (SDES)
- Stress-Blended Eddy Simulation (SBES)

The Kato-Launder formulation is enabled by default only for the Transition SST model.

Both modifications for the production term can be accessed in the Viscous Model panel. For more details, see Including Production Limiters for Two-Equation Models in the *Fluent User's Guide*. The Kato-Launder formulation and production limiter can be applied simultaneously, as is done for the Transition SST model by default.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

4.22. Turbulence Damping

In multiphase flows, a high velocity gradient at the interface between two fluids results in high turbulence generation, in both phases. Hence, turbulence damping is required in the interfacial area to model such flows correctly.

Important:

Turbulence damping is available only with the k-ω and k-ε models.

For the k-ω model, the following source term is added to the ω-equation [161] (p. 1062):

$$S_i = A_i \Delta n \beta \rho_i \left(\frac{B 6 \mu_i}{\beta \rho_i \Delta n^2} \right)^2 \tag{4.443}$$

where

A_i = Interfacial area density for phase i

Δn = Cell height normal to interface

β = k-ω model closure coefficient of destruction term, which is equal to 0.075

B = Damping factor

μ_i = Viscosity of phase i

ρ_i = Density of phase i

For the k-ε model, the following source term is added to the ε-equation:

$$S_{\varepsilon,i} = \frac{A_i \Delta n C_\mu \rho_i}{k_i} \left(\frac{6 B \mu_i k_i}{\rho_i \Delta n^2} \right)^2 \tag{4.444}$$

where

k_i = turbulent kinetic energy of phase i

C_μ = model constant equal to 0.09 (see Model Constants (p. 50))

The interfacial area density for phase i is calculated as

$$A_i = 2.0 \ \alpha_i \ \left| \nabla \alpha_i \right| \tag{4.445}$$

where

α_i = Volume fraction of phase i

$\left| \nabla \alpha_i \right|$ = Magnitude of gradient of volume fraction

The grid size Δn is calculated internally using grid information. You can specify the damping factor B in the **Viscous Model** dialog box. The default value for the damping factor is 10.

Turbulence damping is available with the VOF and Mixture models. Note that it is also available with the Eulerian multiphase model when using the multi-fluid VOF model. To use this option, refer to Turbulence Damping.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

If the Eulerian multiphase model is enabled, you can specify a turbulence multiphase model. If the **Per Phase** turbulence model is used, then the source term is added to the ω or ε equation of each phase. If the VOF or mixture model is enabled, or the Eulerian multiphase model is used with the **Mixture** turbulence model, the term is summed over all the phases and added as a source term to the mixture level ω or ε equation. For information about turbulence multiphase modeling, see Modeling Turbulence.

4.23. Definition of Turbulence Scales

This section outlines the definitions of the turbulent velocity scale v_t and turbulent time scale τ which are provided by the different turbulence models for use in other physical models. The turbulent length scale, l, is then calculated based on the turbulent velocity and time scale in the following way:

$$l = v_t \tau \tag{4.446}$$

The velocity scale v_t is defined as the velocity scale of the modeled turbulence (v_m) limited by the Kolmogorov microscale (v_K):

$$v_t = \max(v_m, v_K) \tag{4.447}$$

In a similar way, the turbulent time scale τ can be defined:

$$\tau = \max(\tau_m, \tau_K) \tag{4.448}$$

4.23.1. RANS and Hybrid (SAS, DES, and SDES) Turbulence Models

For RANS turbulence models and hybrid models (that is, Scale-Adaptive Simulation, Detached Eddy Simulation, and Shielded Detached Eddy Simulation), the velocity scales v_m and v_K are defined as:

$$v_m = \sqrt{k} \tag{4.449}$$

$$v_K = \left(\frac{\varepsilon \mu}{\rho}\right)^{0.25} \tag{4.450}$$

where k is the turbulence kinetic energy, ε is the dissipation rate, μ is the molecular viscosity, and ρ is the density.

The time scales τ_m and τ_K are defined in the following way:

$$\tau_m = \frac{k}{\varepsilon} \tag{4.451}$$

$$\tau_K = \sqrt{\frac{\mu}{\rho \varepsilon}} \tag{4.452}$$

In turbulence models based on the ω-equation, the dissipation rate is calculated using the relation

$$\varepsilon = \beta^* \omega k \tag{4.453}$$

where β^* is a model coefficient.

Using Reynolds stress models, the turbulence kinetic energy is defined as the trace of the Reynolds stress tensor:

$$k = \frac{1}{2} \overline{u_i' u_i'} \tag{4.454}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

4.23.2. Large Eddy Simulation (LES) Models

For LES models, the velocity scales are estimated based on an eddy viscosity μ_t and LES grid length scale l_{LES}:

$$\text{v}_\text{m}=\frac{\mu_t}{\rho l_{\text{LES}}} \tag{4.455}$$

$$\text{v}_\text{K}=\left(\frac{\mu \text{v}_\text{m}^3}{\rho l_{\text{LES}}}\right)^{0.25} \tag{4.456}$$

The time scales τ_m and τ_K are defined in the following way

$$\tau_\text{m}=\frac{l_{\text{LES}}}{\text{v}_\text{m}} \tag{4.457}$$

$$\tau_\text{K}=\sqrt{\frac{\mu l_{\text{LES}}}{\rho \text{v}_\text{m}^3}} \tag{4.458}$$

The definition of the LES grid length scale l_{LES} depends on the specified model. Using the Smagorinsky-Lilly, WALE, or one of the Algebraic Wall-Modeled LES models, it is computed as:

$$l_{\text{LES}}=\min(\kappa d, C\Delta) \tag{4.459}$$

where κ is the von Kármán constant, d is the distance to the closest wall, C a model coefficient, and Δ represents the local grid scale based on the cell volume (as shown in the following equation).

$$\Delta=V^{1/3} \tag{4.460}$$

Note that WMLES uses a more complex formulation of Δ in the definition of the eddy viscosity (see Algebraic WMLES Model Formulation (p. 124) for more details).

In dynamic LES models (that is, the dynamic Smagorinsky-Lilly model and the dynamic kinetic energy subgrid-scale model), the LES grid length scale l_{LES} is estimated based on the dynamic procedure.

4.23.3. Stress-Blended Eddy Simulation (SBES) Model

For the SBES model, the RANS and LES scales (as defined in RANS and Hybrid (SAS, DES, and SDES) Turbulence Models (p. 168) and Large Eddy Simulation (LES) Models (p. 169), respectively) are blended using the f_{SDES} function:

$$\text{v}_\text{t}=\text{v}_{\text{t,RANS}} f_{\text{SDES}}+\text{v}_{\text{t,LES}}\left(1-f_{\text{SDES}}\right) \tag{4.461}$$

$$\tau=\tau_{\text{RANS}} f_{\text{SDES}}+\tau_{\text{LES}}\left(1-f_{\text{SDES}}\right) \tag{4.462}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information
of ANSYS, Inc. and its subsidiaries and affiliates.

Chapter 5: Heat Transfer

This chapter describes the theory behind heat transfer in Ansys Fluent. Information is provided in the following sections:

5.1. Introduction

5.2. Modeling Conductive and Convective Heat Transfer

5.3. Modeling Radiation

For more information about using heat transfer in Ansys Fluent, see Modeling Thermal Energy in the User's Guide.

5.1. Introduction

The flow of thermal energy from matter occupying one region in space to matter occupying a different region in space is known as heat transfer. Heat transfer can occur by three main methods: conduction, convection, and radiation. Physical models involving conduction and/or convection only are the simplest (Modeling Conductive and Convective Heat Transfer (p. 171)), while buoyancy-driven flow or natural convection (Natural Convection and Buoyancy-Driven Flows Theory (p. 176)), and radiation models (Modeling Radiation (p. 183)) are more complex. Depending on your problem, Ansys Fluent will solve a variation of the energy equation that takes into account the heat transfer methods you have specified. Ansys Fluent is also able to predict heat transfer in periodically repeating geometries (Modeling Periodic Heat Transfer in the User's Guide), therefore greatly reducing the required computational effort in certain cases.

For more information about using heat transfer models in Ansys Fluent, see Modeling Conductive and Convective Heat Transfer, Modeling Radiation, and Modeling Periodic Heat Transfer in the User's Guide.

5.2. Modeling Conductive and Convective Heat Transfer

Ansys Fluent allows you to include heat transfer within the fluid and/or solid regions in your model. Problems ranging from thermal mixing within a fluid to conduction in composite solids can therefore be handled by Ansys Fluent. When your Ansys Fluent model includes heat transfer you must enable the relevant physical models, supply thermal boundary conditions, and enter material properties that govern heat transfer and/or vary with temperature as part of the setup. For information about setting up and using heat transfer models in your Ansys Fluent model, see Modeling Thermal Energy in the User's Guide.

Information about heat transfer theory is presented in the following subsections.

5.2.1. Heat Transfer Theory

5.2.2. Natural Convection and Buoyancy-Driven Flows Theory

5.2.3. The Two-Temperature Model for Hypersonic Flows

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

5.2.1. Heat Transfer Theory

5.2.1.1. The Energy Equation

Ansys Fluent solves the energy equation in the following form:

$$\frac{\partial}{\partial t}\left(\rho\left(e+\frac{v^2}{2}\right)\right)+\nabla\cdot\left(\rho\vec{v}\left(h+\frac{v^2}{2}\right)\right)=\nabla\cdot\left(k_{eff}\nabla T-\sum_j h_j\vec{J}_j+\bar{\bar{\tau}}_{eff}\cdot\vec{v}\right)+S_h \tag{5.1}$$

where k_{eff} is the effective conductivity ($k+k_t$), where k_t is the turbulent thermal conductivity, defined according to the turbulence model being used, and $\vec{J}_j$ is the diffusion flux of species j. The first three terms on the right-hand side of Equation 5.1 (p. 172) represent energy transfer due to conduction, species diffusion, and viscous dissipation, respectively. S_h includes volumetric heat sources that you have defined and the heat generation rate from chemical reactions shown in Equation 5.11 (p. 174). However, this reaction source does not apply for the total enthalpy equation (see Energy Sources Due to Reaction (p. 174) for details).

In Equation 5.1 (p. 172),

The enthalpy h is defined for ideal gases as

$$h=\sum_j Y_j h_j \tag{5.2}$$

and for incompressible materials includes the contribution from pressure work

$$h=\sum_j Y_j h_j+\frac{p}{\rho} \tag{5.3}$$

In Equation 5.2 (p. 172) and Equation 5.3 (p. 172), Y_j is the mass fraction of species j and the sensible heat of species h_j is the part of enthalpy that includes only changes in the enthalpy due to specific heat

$$h_j=\int_{T_{ref}}^{T} c_{p,j}dT \tag{5.4}$$

The value used for T_{ref} in the sensible enthalpy calculation depends on the solver and models in use. For the pressure-based solver T_{ref} is 298.15 K except for PDF models in which case T_{ref} is a user input for the species. For the density-based solver T_{ref} is 0 K except when modeling species transport with reactions in which case T_{ref} is a user input for the species.

The internal energy e_q is defined uniformly for compressible and incompressible materials as

$$e=h-\frac{p_{op}+p}{\rho} \tag{5.5}$$

In the above formulas p and p_{op} are gauge and operating pressure, respectively. Such definitions of enthalpy and internal energy accommodate an incompressible ideal gas in the common formulation:

$$h=c_v T+\frac{p_{abs}}{\rho}=c_v T+R_g T+\frac{p}{\rho}=c_p T+\frac{p}{\rho} \tag{5.6}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

5.2.1.2. The Energy Equation in Moving Reference Frames

The energy equation is solved in moving (relative) frames of reference. In moving frames of reference, the energy transport equation uses rothalpy as a conservative quantity. See Equation 2.6 (p. 22) for the energy equation in moving frames of reference.

5.2.1.3. The Energy Equation for the Non-Premixed Combustion Model

When the non-adiabatic non-premixed combustion model is enabled, Ansys Fluent solves the total enthalpy form of the energy equation:

$$\frac{\partial}{\partial t}\left(\rho H\right)+\nabla\cdot\left(\rho\vec{v}H\right)=\nabla\cdot\left(\frac{k_{eff}}{C_p}\nabla H\right)+S_h \tag{5.7}$$

Under the assumption that the Lewis number (Le) = 1, the conduction and species diffusion terms combine to give the first term on the right-hand side of the above equation while the contribution from viscous dissipation appears in the non-conservative form as the second term. The total enthalpy H is defined as

$$H=\sum_j Y_j H_j \tag{5.8}$$

where Y_j is the mass fraction of species j and

$$H_j=\int_{T_{ref,j}}^{T} c_{p,j}dT+h_j^0\left(T_{ref,j}\right) \tag{5.9}$$

$h_j^0\left(T_{ref,j}\right)$ is the formation enthalpy of species j at the reference temperature $T_{ref,j}$.

5.2.1.4. Inclusion of Pressure Work and Kinetic Energy Terms

Equation 5.1 (p. 172) includes pressure work and kinetic energy terms, which are often negligible in incompressible flows. For this reason, the pressure-based solver by default does not include the pressure work or kinetic energy when you are solving incompressible flow. If you want to include these terms, use the `define/models/energy?` text command. When asked to `include pressure work in energy equation?` and `include kinetic energy in energy equation?`, respond by entering `yes` in the console window.

Pressure work and kinetic energy are *always* automatically accounted for when you are:

- modeling compressible flow
- modeling incompressible flow with viscous dissipation and the pressure based solver
- using the density-based solver

5.2.1.5. Inclusion of the Viscous Dissipation Terms

Equation 5.1 (p. 172) and Equation 5.7 (p. 173) describe the thermal energy created by viscous shear in the flow.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

When the pressure-based solver is used, Ansys Fluent's default form of the energy equation does not include them (because viscous heating is often negligible). Viscous heating will be important when the Brinkman number, Br, approaches or exceeds unity, where

$$Br = \frac{\mu U_e^2}{k \Delta T} \tag{5.10}$$

ΔT represents the temperature difference in the system and U_e represents the characteristic velocity of the system.

When your problem requires inclusion of the viscous dissipation terms and you are using the pressure-based solver, you should enable the terms using the **Viscous Heating** option in the Viscous Model Dialog Box. Compressible flows typically have $Br \geq 1$. Note, however, that when the pressure-based solver is used, Ansys Fluent does not automatically enable the viscous dissipation if you have defined a compressible flow model.

When the density-based solver is used, the viscous dissipation terms are *always* included when the energy equation is solved.

5.2.1.6. Inclusion of the Species Diffusion Term

Equation 5.1 (p. 172) and Equation 5.7 (p. 173) both include the effect of enthalpy transport due to species diffusion.

When the pressure-based solver is used, the term

$$\nabla \cdot \left(\sum_j h_j \vec{J}_j \right)$$

is included in Equation 5.1 (p. 172) by default. If you do not want to include it, you can disable the **Diffusion Energy Source** option in the Species Model Dialog Box.

When the non-adiabatic non-premixed combustion model is being used, this term does not explicitly appear in the energy equation, because it is included in the first term on the right-hand side of Equation 5.7 (p. 173).

When the density-based solver is used, this term is *always* included in the energy equation.

5.2.1.7. Energy Sources Due to Reaction

Sources of energy, S_h, in Equation 5.1 (p. 172) include the source of energy due to chemical reaction:

$$S_{h,rxn} = -\sum_j \frac{h_j^0}{M_j} \mathfrak{R}_j \tag{5.11}$$

where h_j^0 is the enthalpy of formation of species j, M_j is the molecular weight of species j, and $\mathfrak{R}_j$ is the volumetric rate of creation of species j.

In the energy equation used for non-adiabatic non-premixed combustion (Equation 5.7 (p. 173)), the heat of formation is included in the definition of enthalpy (see Equation 5.8 (p. 173)), so reaction sources of energy are not included in S_h.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

5.2.1.8. Energy Sources Due To Radiation

When one of the radiation models is being used, S_h in Equation 5.1 (p. 172) or Equation 5.7 (p. 173) also includes radiation source terms. For details, see Modeling Radiation (p. 183).

5.2.1.9. Energy Source Due To Joule Heating

When the electric potential equation Equation 19.1 (p. 858) is being solved, S_h in Equation 5.1 (p. 172) also includes Joule heating source terms.

5.2.1.10. Interphase Energy Sources

It should be noted that the energy sources, S_h, also include heat transfer between the continuous and the discrete phase. This is further discussed in Coupling Between the Discrete and Continuous Phases (p. 583).

5.2.1.11. Energy Equation in Solid Regions

In solid regions, the energy transport equation used by Ansys Fluent has the following form:

$$\frac{\partial}{\partial t}(\rho h)+\nabla\cdot(\vec{v}\rho h)=\nabla\cdot(k\nabla T)+S_h \tag{5.12}$$

where

ρ = density

h = sensible enthalpy, $\int_{T_{ref}}^{T} c_p dT$

k = conductivity

T = temperature

S_h = volumetric heat source

The second term on the left-hand side of Equation 5.12 (p. 175) represents convective energy transfer due to rotational or translational motion of the solids. The velocity field $\vec{v}$ is computed from the motion specified for the solid zone. (For details, see Solid Conditions in the User's Guide). The terms on the right-hand side of Equation 5.12 (p. 175) are the heat flux due to conduction and volumetric heat sources within the solid, respectively.

5.2.1.12. Anisotropic Conductivity in Solids

Ansys Fluent can solve the conduction equation in solid zones and shells with the thermal conductivity specified as a matrix. The heat flux vector is written as:

$$q_i=k_{ij}\left(\frac{\partial T}{\partial x_j}\right) \tag{5.13}$$

where k_{ij} is the thermal conductivity tensor, by default in global coordinates.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$k_{ij}=k_{glob}=\begin{bmatrix} k_{xx} & k_{xy} & k_{xz} \\ k_{yx} & k_{yy} & k_{yz} \\ k_{zx} & k_{zy} & k_{zz} \end{bmatrix} \tag{5.14}$$

k_{glob} can also be calculated as:

$$k_{glob}=Ak_{loc}A^{T} \tag{5.15}$$

where k_{loc} is user specified local thermal conductivities.

$$k_{loc}=\begin{bmatrix} k_1 & 0 & 0 \\ 0 & k_2 & 0 \\ 0 & 0 & k_3 \end{bmatrix} \tag{5.16}$$

A is the transformation matrix and has three principal directions for each cell of a mesh.

$$A=\begin{bmatrix} e_{xx} & e_{xy} & e_{xz} \\ e_{yx} & e_{yy} & e_{yz} \\ e_{zx} & e_{zy} & e_{zz} \end{bmatrix} \tag{5.17}$$

A matrix is provided in current orthotropic material panel along with k_{loc}. See Anisotropic Thermal Conductivity for Solids in the User's Guide for details on specifying anisotropic conductivity for solid materials.

A transformation matrix has constant directions, and therefore would not follow curved geometry. For curved geometries, the transformation matrix is calculated for each cell to appropriately calculate k_{glob}. For details on the specifying anisotropic conductivity for curved geometry, see Anisotropic Thermal Conductivity with Curvilinear Coordinate System (CCS) in the *Fluent User's Guide*.

5.2.1.13. Diffusion at Inlets

The net transport of energy at inlets consists of both the convection and diffusion components. The convection component is fixed by the inlet temperature specified by you. The diffusion component, however, depends on the gradient of the computed temperature field. Thus the diffusion component (and therefore the net inlet transport) is not specified a priori.

In some cases, you may want to specify the net inlet transport of energy rather than the inlet temperature. If you are using the pressure-based solver, you can do this by disabling inlet energy diffusion. By default, Ansys Fluent includes the diffusion flux of energy at inlets. To turn off inlet diffusion, use the `define/models/energy?` text command and respond `no` when asked to `Include diffusion at inlets?`

Inlet diffusion cannot be turned off if you are using the density-based solver.

5.2.2. Natural Convection and Buoyancy-Driven Flows Theory

When heat is added to a fluid and the fluid density varies with temperature, a flow can be induced due to the force of gravity acting on the density variations. Such buoyancy-driven flows are termed natural-convection (or mixed-convection) flows and can be modeled by Ansys Fluent.

The importance of buoyancy forces in a mixed convection flow can be measured by the ratio of the Grashof and Reynolds numbers:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$\frac{Gr}{Re^2} = \frac{g\beta \Delta TL}{v^2} \tag{5.18}$$

When this number approaches or exceeds unity, you should expect strong buoyancy contributions to the flow. Conversely, if it is very small, buoyancy forces may be ignored in your simulation. In pure natural convection, the strength of the buoyancy-induced flow is measured by the Rayleigh number:

$$Ra = \frac{g\beta \Delta TL^3 \rho}{\mu \alpha} \tag{5.19}$$

where β is the thermal expansion coefficient:

$$\beta = -\frac{1}{\rho}\left(\frac{\partial \rho}{\partial T}\right)_p \tag{5.20}$$

and α is the thermal diffusivity:

$$\alpha = \frac{k}{\rho c_p} \tag{5.21}$$

Rayleigh numbers less than 10^8 indicate a buoyancy-induced laminar flow, with transition to turbulence occurring over the range of $10^8 < Ra < 10^{10}$.

5.2.3. The Two-Temperature Model for Hypersonic Flows

When using the density-based solver, a two-temperature model is available for simulating the thermal non-equilibrium phenomena in hypersonic flows. It models the energy relaxation process in the flow and provides a better prediction of the flow fields than the one-temperature model.

The degree of thermal non-equilibrium is typically measured by a suitable Damkohler number, expressed as the ratio between the characteristic time for a fluid element to travel a characteristic length and the characteristic time for the energy modes to reach equilibrium. When the flow speed is relatively low, the Damkohler number is much larger than one, which means the number of molecular collisions is sufficient for the flow to reach the local thermodynamic equilibrium state. For the hypersonic flows, the Damkohler number is typical of the order of unity. This indicates that the fluid element does not reside at one location long enough to bring the local thermodynamic state to equilibrium, and therefore the flow is in thermal non-equilibrium. To properly address this effect, it is necessary to use a two-temperature model that assumes: the translational and rotational energy modes of species are in equilibrium with one temperature; and the vibrational and electronic energy modes of species are in equilibrium with another temperature. A system of conservation equations is solved that includes the Navier-Stokes equations and one additional transport equation that models the conservation of vibrational-electronic energy.

Note:

Additional feature license is required to enable the use of this model. Please contact your Ansys representative to check for the availability of the license increment.

Information about the equations is provided in the following:

5.2.3.1. Energy Equations

5.2.3.2. Equation of States

5.2.3.3. Specific Heat

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

5.2.3.4. Transport Properties

5.2.3.5. Limitations

5.2.3.1. Energy Equations

With the two-temperature model, the governing equation for the vibrational-electronic energy [211] (p. 1065) is

$$\frac{\partial \rho e_{ve}}{\partial t} + \nabla \cdot (\rho e_{ve} \mathbf{V}) = -\nabla \cdot \mathbf{q}_{ve} - \sum_i \nabla \cdot (e_{ve,i} J_i) + S^{tv} + S^{cv} + S^{he} + S^{ei} + S^{epg} \tag{5.22}$$

where e_{ve} is the vibrational-electronic energy per unit mass. This equation is solved segregated from the Navier-Stokes equations (Equation 24.74 (p. 991)) in the density-based solver.

Since the two-temperature model introduces T_{tr} and T_{ve} as the translational-rotational and vibrational-electronic temperature, respectively, the heat flux term in the original energy equation Equation 24.75 (p. 991) is adjusted to:

$$q = k_{tr} \nabla T_{tr} + k_{ve} \nabla T_{ve} \tag{5.23}$$

$\sum_i \nabla \cdot e_{ve,i} J_i$ models the change in vibrational-electronic energy due to species diffusion. J_i is the diffusion flux of species $_i$, defined as:

$$J_i = -\rho D_{i,m} \nabla Y_I \tag{5.24}$$

which is the same definition used in Equation 7.2 (p. 238), but neglecting thermal diffusion. If ions and free electrons are present, the ambipolar diffusion assumption is used to calculate the diffusion flux of electrons:

$$J_e = M_e \sum_{s \neq e} \frac{J_s C_s}{M_s} \tag{5.25}$$

where M_s is the species molecular weight and C_s is the species charge.

The source term S^{tv} in Equation 5.22 (p. 178) models the relaxation between translational and vibrational modes. The relaxation source term is modeled using the Landau-Teller theory as

$$S^{tv} = \sum \rho_s \frac{e_{v,s}(T_{tr}) - e_{v,s}(T_{ve})}{\tau_s} \tag{5.26}$$

where $e_{v,s}$ is the species vibrational energy and τ_s is the relaxation time modeled by Millikan and White's curve fits and Park's correction [571] (p. 1086).

For triatomic molecules, this approach assumes that the energy transfer between translational and vibrational modes is determined by the fastest relaxation time, corresponding to the lowest characteristic vibrational temperature.

$$\tau_{sr} = \frac{101325}{p} \exp[A_{sr}(T^{-\frac{1}{3}} - B_{sr}) - 18.42] \tag{5.27}$$

where the vibrational constants A_{sr} and B_{sr} are computed from Millikan-White's formula.

$$A_{sr} = 1.16 \times 10^{-3} \mu_{sr}^{1/2} \left(\min_i \theta_{v,s,i} \right)^{4/3}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$B_{sr}=0.015\mu_{sr}^{1/4}$$

$$\mu_{sr}=\frac{M_s M_r}{M_s+M_r}$$

The relaxation time defined in Millikan and White's expression does not represent experimental data for certain collisions. To address this issue, Park [766] (p. 1097)[767] (p. 1097) provides coefficients A_{sr} and B_{sr} derived from curve fitting to experimental data for those collision cases. However, Millikan and White's method is still applied for other collisions. In the two-temperature model, Park's model is set as the default option. When selecting Park's model, the vibrational constants of those collisions will be automatically defined. You can modify the translational-vibrational energy relaxation model settings by using the following console command:

```
/define/models/two-temperature/translational-vibrational-energy-relax-
ation
```

The source term S^{cv} models the vibrational-electronic energy changes due to chemical reactions. The non-preferential model is adopted, assuming that the molecules are created or destroyed at the average vibrational energy level.

$$S^{cv}=\sum_{molecules} R_i e_{ve,i} \tag{5.28}$$

where R_i is the net source of species $_i$ due to chemical reactions, defined in Equation 7.19 (p. 243).

The source term S^{he} in Equation 5.22 (p. 178) models the energy transfer between heavy particles and electrons.

$$S^{he}=3R_u\rho_e(T_{tr}-T_{ve})\sqrt{\frac{8R_uT_{ve}}{\pi M_e}}\sum_{r\neq e}\frac{\rho_r N_a}{M_r^2}\sigma_{er} \tag{5.29}$$

where N_a is Avogadro's constant, and σ_{er} is the collision cross-section between electrons and heavy particles given in [571] (p. 1086).

The source term S^{ei} models the energy removed from free electrons during impact ionization reactions.

$$S^{ei}=M_{N^+}\hat{R}_{N^+}\hat{I}_N+M_{O^+}\hat{R}_{O^+}\hat{I}_O \tag{5.30}$$

$\hat{R}_{N^+}$ is the molar change rate of N^+ of the impact ionization reaction involving N, and $\hat{R}_{O^+}$ is the molar change rate of O^+ of the impact ionization reaction involving O. $\hat{I}_N$ and $\hat{I}_O$ are 1/3 of the first ionization energy of N and O given in [211] (p. 1065).

S^{epg} models the work on electron by the induced electric field.

$$S^{epg}=-p_e\nabla\cdot\mathbf{V} \tag{5.31}$$

where p_e is the electron pressure.

5.2.3.2. Equation of States

Each species is assumed to follow the ideal gas law.

$$p_s = \begin{cases} \rho_s R_s T_{ve} & electrons \\ \rho_s R_s T_{tr} & heavy\ particles \end{cases} \tag{5.32}$$

Dalton's partial pressure law is used to obtain the mixture pressure.

$$p = \sum_s p_s \tag{5.33}$$

5.2.3.3. Specific Heat

When the two-temperature model is enabled, you have the option of computing the specific heat using the nasa-9-piecewise-polynomial or the Boltzmann-kinetic-theory.

5.2.3.3.1. NASA-9-Piecewise-Polynomial Functions for the Two-Temperature Model

The default method used to model thermodynamic relations is the nasa-9-piecewise-polynomial [211] (p. 1065). A piecewise polynomial is defined for the specific heat as a function of temperature.

$$C_p(T) = A_1 T^{-2} + A_2 T^{-1} + A_3 + A_4 T + A_5 T^2 + A_6 T^3 + A_7 T^4 \tag{5.34}$$

Equation 5.34 (p. 180) is only valid in thermodynamics equilibrium ($T_{tr} = T_{ve}$), however, with the assumption that the translational and rotational energy modes are fully excited, the vibrational and electronic specific heat can be evaluated from Equation 5.34 (p. 180). The translational and rotational specific heat can be assumed as constant.

$$C_{p,tr} = \begin{cases} \frac{7}{2} R_s & molecules \\ \frac{5}{2} R_s & atoms\ or\ electrons \end{cases} \tag{5.35}$$

where R_s is the gas constant for species s. Since $C_{p,tr}$ is independent of temperature, the specific heat can be written as:

$$C_p(T_{ve}) = C_{p,tr} + C_{p,ve}(T_{ve}) \tag{5.36}$$

Substituting $T = T_{ve}$ into Equation 5.34 (p. 180) and comparing it with equation Equation 5.36 (p. 180), the vibrational and electronic specific heat can be expressed as

$$C_{p,ve}(T_{ve}) = C_p(T_{ve}) - C_{p,tr} \tag{5.37}$$

5.2.3.3.2. Boltzmann Energy Distribution Theory

From statistic mechanics, the internal energy can be stored as translational, rotational, vibrational and electronic modes. The species internal energy is

$$e_{int,s} = e_{t,s}(T_{tr}) + e_{r,s}(T_{tr}) + e_{v,s}(T_{ve}) + e_{e,s}(T_{ve}) + h_{0,s} \tag{5.38}$$

where $e_{int,s}$ is the species internal energy; $e_{t,s}$, $e_{r,s}$, $e_{v,s}$, and $e_{e,s}$ are species translational, rotational, vibrational, and electronic energy, respectively; and $h_{0,s}$ is the species heat of formation. The rigid-rotor harmonic-oscillator approximation is adopted to calculate the energy modes [571] (p. 1086).

The number of vibrational modes depends on the number of atoms in a molecule. A diatomic molecule contains one vibrational mode, while a triatomic molecule can contain three or four vibrational modes.

To account for multiple vibrational modes, the vibrational energy is defined as

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$e_{v,s}=\sum_i g_{s,i}R_s\frac{\theta_{v,s,i}}{e^{\theta_{v,s,i}/T_{ve}}-1} \tag{5.39}$$

where $\theta_{v,s,i}$ and $g_{s,i}$ are the characteristic vibrational temperature and the degeneracy of vibrational mode i and species s, respectively.

The species specific heat is given by

$$c_{p,s}=c_{p,t,s}+c_{p,r,s}+c_{p,v,s}+c_{p,e,s} \tag{5.40}$$

where $c_{p,s}$ is the species specific heat, $c_{p,t,s}$, $c_{p,r,s}$, $c_{p,v,s}$ and $c_{p,e,s}$ are species translational, rotational, vibrational, and electronic specific heat. The translational and rotational specific heat is given in Equation 5.35 (p. 180).

The vibrational specific heat can be calculated as

$$c_{p,v,s}=\sum_i g_{s,i}R_s\frac{\left(\theta_{v,s,i}/T_{ve}\right)^2 e^{\theta_{v,s,i}/T_{ve}}}{\left[e^{\theta_{v,s,i}/T_{ve}}-1\right]^2} \tag{5.41}$$

The electronic mode is not modeled in the Boltzmann-kinetic-theory method.

5.2.3.4. Transport Properties

Two approaches for transport properties are introduced. The Blottner/Eucken/Constant Lewis methods are suitable for relatively low-speed flows where the freestream speed is less than 7km/h, and the maximal temperature is lower than 10,000K. The Gupta mixing law with collision cross-section data is appropriate for weakly-ionized flows with higher flow field temperatures.

5.2.3.4.1. Blottner/Eucken/Constant Lewis

The Blottner's curve fit is adopted to calculate the species viscosity:

$$\mu_s=0.1\exp\left[\left(A_s\ln T_{tr}+B_s\right)\ln T_{tr}+C_s\right] \tag{5.42}$$

The Eucken's relation is used to calculate the translational-rotational and vibrational-electronic contributions to the species thermal conductivities:

$$k_{tr,s}=\mu_s\left(\frac{5}{2}C_{v,s}^t+C_{v,s}^r\right) \tag{5.43}$$

$$k_{ve,s}=\mu_s C_{v,s}^{ve} \tag{5.44}$$

To model the diffusivity coefficient $D_{i,m}$, a constant Lewis number assumption is adopted.

$$D_{i,m}=\frac{Lek_{tr}}{\rho C_p^{tr}} \tag{5.45}$$

where Le is the Lewis number (1.4 by default), k_{tr} is the translation-rotational thermal conductivity, and C_p^{tr} is the translational-rotational specific heat at constant pressure. When ions and free electrons are present, the ambipolar diffusion constraint is adopted to maintain zero current density. The diffusion coefficients of ions and electrons are assumed to be doubled [96] (p. 1058).

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

5.2.3.4.2. Gupta's Mixing Law

The Gupta's mixing law and collision cross-section data [571] (p. 1086) are used to calculate the transport coefficients and is adequate for weakly-ionized flows.

The mixture viscosity is

$$\mu=\sum_{s\neq e}\frac{m_s\gamma_s}{\sum_{r\neq e}\gamma_r\Delta_{sr}^{(2)}(T_{tr})+\gamma_e\Delta_{se}^{(2)}(T_{ve})}+\frac{m_e\gamma_e}{\sum_r\gamma_r\Delta_{er}^{(2)}(T_{ve})} \tag{5.46}$$

where γ_s is the molar concentration and m_s is the mass of species.

The mixture translational thermal conductivity is

$$k_t=\frac{15}{4}k_{B,SI}\sum_{s\neq e}\frac{\gamma_s}{\sum_{r\neq e}\alpha_{sr}\gamma_r\Delta_{sr}^{(2)}(T_{tr})+3.54\gamma_e\Delta_{se}^{(2)}(T_{ve})} \tag{5.47}$$

where $k_{B,SI}$ is the Boltzmann constant in SI units.

The mixture rotational thermal conductivity is

$$k_r=k_{B,SI}\sum_{s=molecule}\frac{\gamma_s}{\sum_{r\neq e}\gamma_r\Delta_{sr}^{(1)}(T_{tr})+\gamma_e\Delta_{se}^{(1)}(T_{ve})} \tag{5.48}$$

The mixture vibrational-electronic thermal conductivity is

$$k_{ve}=k_{B,SI}\frac{C_{v,ve}}{R}\sum_{s=molecule}\frac{\gamma_s}{\sum_{r\neq e}\gamma_r\Delta_{sr}^{(1)}(T_{tr})+\gamma_e\Delta_{se}^{(1)}(T_{ve})} \tag{5.49}$$

The electron thermal conductivity is

$$k_e=\frac{15}{4}k_{B,SI}\frac{\gamma_e}{\sum_r 1.45\gamma_r\Delta_{er}^{(2)}(T_{ve})} \tag{5.50}$$

The species diffusion coefficient [567] (p. 1086) is

$$D_s=\begin{cases}\frac{1-x_s}{\sum_{r\neq s}(x_r/D_{sr})} & neutrals\\ 2\frac{1-x_s}{\sum_{r\neq s}(x_r/D_{sr})} & ions\\ \frac{\sum_{s=ions}Y_sD_s}{\sum_{s=ions}Y_s} & electrons\end{cases} \tag{5.51}$$

where D_{sr} is the binary diffusion coefficients, defined as:

$$D_{sr}=\begin{cases}\frac{k_{B,SI}T_{tr}}{p\Delta_{sr}^{(1)}(T_{tr})} & s\ and\ r\ are\ heavy\ particles\\ \frac{k_{B,SI}T_{ve}}{p\Delta_{sr}^{(1)}(T_{ve})} & s\ or\ r\ is\ electron\end{cases} \tag{5.52}$$

The collision cross-sections are given by

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$\Delta_{sr}^{(1)}(T)=\frac{8}{3}\left[\frac{2M_sM_r}{\pi R_uT(M_s+M_r)}\right]^{1/2}10^{-20}\pi\Omega_{sr}^{(1,1)}(T)$$
$$\Delta_{sr}^{(2)}(T)=\frac{16}{5}\left[\frac{2M_sM_r}{\pi R_uT(M_s+M_r)}\right]^{1/2}10^{-20}\pi\Omega_{sr}^{(2,2)}(T) \quad (5.53)$$

where $\Omega_{sr}^{(1,1)}$ is the diffusion collision integral and $\Omega_{sr}^{(2,2)}$ is the viscosity collision integral. For neutral-involved collisions, the collision integral can be fitted to [571] (p. 1086).

$$\pi\Omega_{sr}^{(n,n)}(T)=DT^{\left[A(lnT)^2+B\ln T+C\right]} \quad (5.54)$$

For collisions between charged particles, the interactions are represented by shielded Coulomb potentials:

$$\pi\Omega_{sr}^{(n,n)}(T)=5\times10^{15}\pi\left(\lambda_D/\mathrm{T}^*\right)^2\ln\left\{D_nT^*\left[1-C_n\exp(-c_nT^*)\right]+1\right\} \quad (5.55)$$

where λ_D is the Debye shielding length in centimeters, and T^* is the reduced temperature defined in [714] (p. 1094). The update-to-date coefficients have been implemented for air mixtures in the Fluent material database.

5.2.3.5. Limitations

The following limitations apply to the two-temperature model:

- It is only available for the density-based solver.
- Non-zero surface heat flux is not supported.
- The electronic energy mode is not modeled in boltzmann-kinetic-theory method.
- The coefficients for Blottner's method and the characteristic vibrational temperature are provided for Air, N_2, O_2, NO, N, O, $N2+$, $O2+$, $NO+$, $N+$, $O+$, e, C, CO_2, CO, and Ar in the Fluent material database. If you want to model other species, you will need to provide the data.
- The collision cross-section coefficients are required for the Gupta mixing law. These coefficients are built-in for the `air-2species-nitrogen`, `air-5species-park93`, `air-11species-gupta`, and `air-11species-park93` mixtures. You will need to provide the coefficients if you want to use other mixtures.
- The species' chemical formula is used to identify electrons or ions. For example, the formula e will be treated as an electron and N+ will be treated as an ion. Therefore, you should not change the species' chemical formula when importing materials.

5.3. Modeling Radiation

Information about radiation modeling theory is presented in the following sections:

5.3.1. Overview and Limitations
5.3.2. Radiative Transfer Equation
5.3.3. P-1 Radiation Model Theory
5.3.4. Rosseland Radiation Model Theory
5.3.5. Discrete Transfer Radiation Model (DTRM) Theory

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

5.3.6. Discrete Ordinates (DO) Radiation Model Theory
5.3.7. Surface-to-Surface (S2S) Radiation Model Theory
5.3.8. Monte Carlo (MC) Radiation Model Theory
5.3.9. Radiation in Combusting Flows
5.3.10. Choosing a Radiation Model

For information about setting up and using radiation models in Ansys Fluent, see Modeling Radiation in the User's Guide.

5.3.1. Overview and Limitations

Ansys Fluent provides six radiation models that allow you to include radiation, with or without a participating medium, in your heat transfer simulations.

Heating or cooling of surfaces due to radiation and/or heat sources or sinks due to radiation within the fluid phase can be included in your model using one of the following radiation models.

- Discrete Transfer Radiation Model (DTRM) [94] (p. 1058), [586] (p. 1087)
- P-1 Radiation Model [106] (p. 1059), [593] (p. 1087)
- Rosseland Radiation Model [593] (p. 1087)
- Surface-to-Surface (S2S) Radiation Model [593] (p. 1087)
- Discrete Ordinates (DO) Radiation Model [110] (p. 1059), [535] (p. 1084)
- Monte Carlo (MC) Radiation Model

In addition to these radiation models, Ansys Fluent also provides a solar load model that allows you to include the effects of solar radiation in your simulation.

Typical applications well suited for simulation using radiative heat transfer include the following:

- radiative heat transfer from flames
- surface-to-surface radiant heating or cooling
- coupled radiation, convection, and/or conduction heat transfer
- radiation through windows in HVAC applications, and cabin heat transfer analysis in automotive applications
- radiation in glass processing, glass fiber drawing, and ceramic processing

You should include radiative heat transfer in your simulation when the radiant heat flux, $Q_{rad}=\sigma\left(T_{max}^4-T_{min}^4\right)$, is large compared to the heat transfer rate due to convection or conduction. Typically this will occur at high temperatures where the fourth-order dependence of the radiative heat flux on temperature implies that radiation will dominate.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

5.3.1.1. Advantages and Limitations of the DTRM

The primary advantages of the DTRM are threefold: it is a relatively simple model, you can increase the accuracy by increasing the number of rays, and it applies to a wide range of optical thicknesses.

You should be aware of the following limitations when using the DTRM in Ansys Fluent:

- DTRM assumes that all surfaces are diffuse. This means that the reflection of incident radiation at the surface is isotropic with respect to the solid angle.
- The effect of scattering is not included.
- The implementation assumes gray radiation.
- Solving a problem with a large number of rays is CPU-intensive.
- DTRM is not compatible with non-conformal interfaces or sliding meshes.
- DTRM is not compatible with parallel processing.

5.3.1.2. Advantages and Limitations of the P-1 Model

The P-1 model has several advantages over the DTRM. For the P-1 model, the RTE (Equation 5.56 (p. 189)) is a diffusion equation, which is easy to solve with little CPU demand. The model includes the effect of scattering. For combustion applications where the optical thickness is large, the P-1 model works reasonably well. In addition, the P-1 model can easily be applied to complicated geometries with curvilinear coordinates.

You should be aware of the following limitations when using the P-1 radiation model:

- The P-1 model assumes that all surfaces are diffuse. This means that the reflection of incident radiation at the surface is isotropic with respect to the solid angle.
- The implementation is restricted to either gray radiation or non-gray radiation using a gray-band model. The non-gray implementation assumes a constant absorption coefficient within each wavelength band. The weighted-sum-of-gray-gases model (WSGGM) cannot be used to specify the absorption coefficient in each band. The non-gray implementation also assumes the spectral emissivity at walls to be constant within each band. Further information about the gray-band model is provided in Advantages and Limitations of the DO Model (p. 186).
- There may be a loss of accuracy, depending on the complexity of the geometry, if the optical thickness is small.
- The P-1 model tends to over-predict radiative fluxes from localized heat sources or sinks.

5.3.1.3. Advantages and Limitations of the Rosseland Model

The Rosseland model has two advantages over the P-1 model. Since it does not solve an extra transport equation for the incident radiation (as the P-1 model does), the Rosseland model is faster than the P-1 model and requires less memory.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

The Rosseland model can be used only for optically thick media. It is recommended for use when the optical thickness exceeds 3. Note also that the Rosseland model is not available when the density-based solver is being used; it is available with the pressure-based solver, only.

5.3.1.4. Advantages and Limitations of the DO Model

The DO model spans the entire range of optical thicknesses, and allows you to solve problems ranging from surface-to-surface radiation to participating radiation in combustion problems. It also allows the solution of radiation at semi-transparent walls. Computational cost is moderate for typical angular discretizations, and memory requirements are modest.

The current implementation is restricted to either gray radiation or non-gray radiation using a gray-band model. Solving a problem with a fine angular discretization may be CPU-intensive.

The non-gray implementation in Ansys Fluent is intended for use with participating media with a spectral absorption coefficient a_λ that varies in a stepwise fashion across spectral bands, but varies smoothly within the band. Glass, for example, displays banded behavior of this type. The current implementation does not model the behavior of gases such as carbon dioxide or water vapor, which absorb and emit energy at distinct wave numbers [444] (p. 1078). The modeling of non-gray gas radiation is still an evolving field. However, some researchers [181] (p. 1063) have used gray-band models to model gas behavior by approximating the absorption coefficients within each band as a constant. The implementation in Ansys Fluent can be used in this fashion if desired.

The non-gray implementation in Ansys Fluent is compatible with all the models with which the gray implementation of the DO model can be used. Thus, it is possible to include scattering, anisotropy, semi-transparent media, and particulate effects. However, the non-gray implementation assumes a constant absorption coefficient within each wavelength band. The weighted-sum-of-gray-gases model (WSGGM) cannot be used to specify the absorption coefficient in each band. The implementation allows the specification of spectral emissivity at walls. The emissivity is assumed to be constant within each band.

You should be aware of the following limitations when using the DO radiation model:

- The DO model is not supported for use with granular (fluid-solid) Eulerian multiphase flows.
- The DO/Energy Coupling method is not supported with multiphase models.
- Non-conformal interfaces are not supported with the accelerated DO model with GPU. For more information on the accelerated DO model, see Accelerating Discrete Ordinates (DO) Radiation Calculations in the *Fluent User's Guide*.

5.3.1.5. Advantages and Limitations of the S2S Model

The surface-to-surface (S2S) radiation model is good for modeling the enclosure radiative transfer without participating media (for example, spacecraft heat rejection systems, solar collector systems, radiative space heaters, and automotive underhood cooling systems). In such cases, the methods for participating radiation may not always be efficient. As compared to the DTRM and the DO radiation models, the S2S model has a much faster time per iteration, although the view factor calculation itself is CPU-intensive. This increased time for view factor calculation will be especially pronounced when the emitting/absorbing surfaces are the polygonal faces of polyhedral cells.

You should be aware of the following limitations when using the S2S radiation model:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

- The S2S model assumes that all surfaces are diffuse.
- The implementation assumes gray radiation.
- The storage and memory requirements increase very rapidly as the number of surface faces increases. This can be minimized by using a cluster of surface faces, although the CPU time will be independent of the number of clusters that are used if the face to face basis is used to calculate the view factors.
- The S2S model cannot be used to model participating radiation problems.
- The S2S model with the Hemicube view factor method cannot be used if your model contains symmetry or periodic boundary conditions.
- The S2S model with Cluster to Cluster and Ray Tracing view factor calculation does not support periodic non-conformal interfaces.
- The S2S model does not support hanging nodes or hanging node adaption on radiating boundary zones.
- The S2S model does not support semi-transparent boundaries.
- View factor files cannot be written on mixed windows-linux platforms when clustering is enabled.

5.3.1.6. Advantages and Limitations of the MC Model

The MC model can solve problems ranging from optically thin (transparent) regions to optically thick (diffusion) regions, like combustion. It allows you to calculate quasi-exact solutions. While it is more accurate compared to other available models, it has a higher computational cost.

You should be aware of the following limitations when using the MC radiation model:

- For the Monte Carlo model, all the physical quantities of interest are calculated as surface or volume averages.
- The following notable features/models are supported with the Monte Carlo model:
  - Supports meshes with non-conformal interfaces based on the one-to-one auto-pairing method, such as meshes generated using the Fault Tolerant Meshing (FTM) workflow.
  - Shell conduction is not at all supported with semi-transparent walls and is not supported with opaque walls with boundary source specified, see Opaque Walls for the MC Model.
  - Symmetry planes and periodic boundaries are treated as specular surfaces when using the Monte Carlo model.
  - Transient.
  - Heat exchanger model.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

  - Euler-Euler Multiphase models (VOF, Mixture, Eulerian models).

    Note:

    The radiation field for multiphase is assumed to be homogeneous, and only solves for a single radiation field (not per phase). Radiation bulk material properties are assumed to be a volume fraction weighted average from the phase values. This makes the solution rather sensitive to the mesh, if more than one phase is transparent. Opaque phases should be modeled with high absorption coefficient.

    For dispersed phases (as distinct from continuous morphology) that have small droplet sizes, scattering effects are not included from such small droplets. In general, the accuracy of the approach will reduce as the dispersed phase size gets smaller in relation to the mesh size.

- The following are currently not supported with the Monte Carlo model:
  - 2D cases.
  - Rotating / moving / overset meshes.
  - CutCell meshes.
  - Hanging nodes in the mesh.
  - Thin walls (for example, as baffles).
  - DPM.
  - Euler-Lagrangian Multiphase models.
  - Porous medium (or NETM model).
  - Combustion PDF transport.
  - Boundary sources of radiation for flow boundaries/openings.
  - Two zones with an interior boundary between them must both participate in radiation.
  - Isotropic radiation flux option is not available for the boundary source of radiation on opaque walls.
  - Opaque walls with boundary source and shell conduction (shell conduction is disabled if you enable both).
  - Semi-transparent walls with shell conduction, see Opaque Walls for the MC Model.
  - Mesh adaption is not supported with Monte Carlo model.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

5.3.2. Radiative Transfer Equation

The radiative transfer equation (RTE) for an absorbing, emitting, and scattering medium at position $\vec{r}$ in the direction $\vec{s}$ is

$$\frac{dI(\vec{r},\vec{s})}{ds}+(a+\sigma_s)I(\vec{r},\vec{s})=an^2\frac{\sigma T^4}{\pi}+\frac{\sigma_s}{4\pi}\int_0^{4\pi}I(\vec{r},\vec{s}')\Phi(\vec{s}\cdot\vec{s}')d\Omega' \tag{5.56}$$

where

$\vec{r}$ = position vector

$\vec{s}$ = direction vector

$\vec{s}'$ = scattering direction vector

s = path length

a = absorption coefficient

n = refractive index

σ_s = scattering coefficient

σ = Stefan-Boltzmann constant ($5.67 \times 10^{-8}\ W/m^2-K^4$)

I = radiation intensity, which depends on position ($\vec{r}$) and direction ($\vec{s}$)

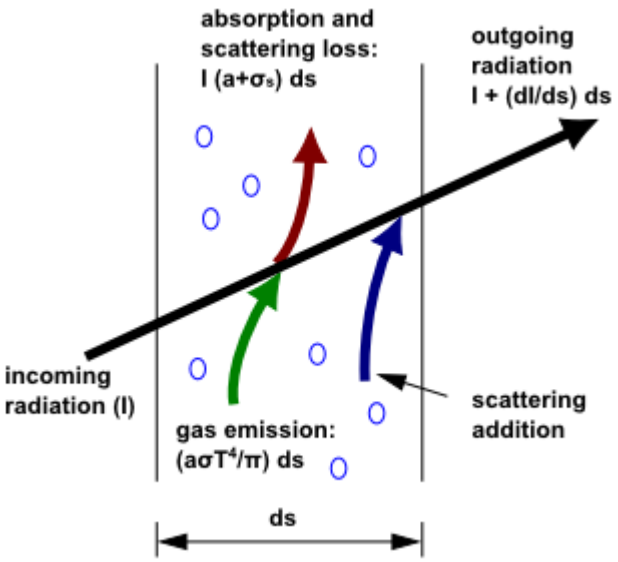
T = local temperature

Φ = phase function

Ω' = solid angle

$(a+\sigma_s)s$ is the optical thickness or opacity of the medium. The refractive index n is important when considering radiation in semi-transparent media. Figure 5.1: Radiative Heat Transfer (p. 189) illustrates the process of radiative heat transfer.

Figure 5.1: Radiative Heat Transfer



The DTRM, P-1, Rosseland, DO, and MC radiation models require the absorption coefficient a as input. a and the scattering coefficient σ_s can be constants, and a can also be a function of local concentrations of H_2O and CO_2, path length, and total pressure. Ansys Fluent provides the weighted-sum-of-gray-gases model (WSGGM) for computation of a variable absorption coefficient. For details, see Radiation in Combusting Flows (p. 218). The DO and MC implementations can model radiation in semi-transparent media. The refractive index n of the medium must be provided as a part of the calculation for this type of problem. The Rosseland model also requires you to enter a refractive index, or use the default value of 1.

The P-1 and DO radiation model is solved for the mixture material. Radiative properties, such as the absorption coefficient, refractive index, and scattering coefficients of the mixture, are computed based on the volume fraction based averaging of the radiative properties of the individual phases.

In simulations that use the porous media model with a radiation model, the contributions of the radiative heat flux and radiative heat source in the energy calculation are scaled by the local porosity. Note that the modeling treatment for the interaction of porous media and radiation in Ansys Fluent is strictly valid only for values of local fluid porosity that are very close or equal to 1.0.

5.3.3. P-1 Radiation Model Theory

The P-1 radiation model is the simplest case of the more general P-N model, which is based on the expansion of the radiation intensity I into an orthogonal series of spherical harmonics [106] (p. 1059), [593] (p. 1087). This section provides details about the equations used in the P-1 model. For information about setting up the model, see Using the Radiation Models in the User's Guide.

5.3.3.1. The P-1 Model Equations

As mentioned above, the P-1 radiation model is the simplest case of the P-N model. When modeling gray radiation, the following equation is obtained for the radiation flux q_r if only four terms in the series are used:

$$q_r = -\frac{1}{3(a+\sigma_s)-C\sigma_s}\nabla G \tag{5.57}$$

where a is the absorption coefficient, σ_s is the scattering coefficient, G is the incident radiation, and C is the linear-anisotropic phase function coefficient, described below. After introducing the parameter

$$\Gamma = \frac{1}{(3(a+\sigma_s)-C\sigma_s)} \tag{5.58}$$

Equation 5.57 (p. 190) simplifies to

$$q_r = -\Gamma\nabla G \tag{5.59}$$

The transport equation for G is

$$\nabla\cdot(\Gamma\nabla G) - aG + 4an^2\sigma T^4 = S_G \tag{5.60}$$

where n is the refractive index of the medium, σ is the Stefan-Boltzmann constant and S_G is a user-defined radiation source. Ansys Fluent solves this equation to determine the local incident radiation when the P-1 model is active.

Combining Equation 5.59 (p. 190) and Equation 5.60 (p. 190) yields the following equation:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$-\nabla \cdot q_r = aG - 4an^2\sigma T^4 \tag{5.61}$$

The expression for $-\nabla \cdot q_r$ can be directly substituted into the energy equation to account for heat sources (or sinks) due to radiation.

Ansys Fluent also allows for the modeling of non-gray radiation using a gray-band model. For non-gray radiation, Equation 5.60 (p. 190) is rewritten as:

$$\nabla \cdot \left(\Gamma_\lambda \nabla G_\lambda\right) - a_\lambda G_\lambda + 4a_\lambda n^2 \sigma T^4 = S_{G\lambda} \tag{5.62}$$

where G_λ is the spectral incident radiation, a_λ is a spectral absorption coefficient, n is the refractive index of the medium, $S_{G\lambda}$ is a user-defined source term, and σ is the Stefan-Boltzmann constant. Γ_λ is defined as:

$$\Gamma_\lambda = \frac{1}{(3(a_\lambda + \sigma_{s\lambda}) - C\sigma_{s\lambda})} \tag{5.63}$$

where $\sigma_{s\lambda}$ is a spectral scattering coefficient and C is the linear anisotropic phase function coefficient.

The spectral black body emission ($G_{b\lambda}$) between wavelength λ_1 and λ_2 is given as

$$G_{b\lambda} = 4[F(0 \rightarrow n\lambda_2 T) - F(0 \rightarrow n\lambda_1 T)]\sigma T^4 \tag{5.64}$$

where $F(0 \rightarrow n\lambda T)$ is the fraction of radiant energy emitted by a black body in the wavelength interval from 0 to λ at temperature T in a medium of refractive index n. λ_2 and λ_1 are the wavelength boundaries of the band.

The spectral radiative flux is calculated as

$$q_\lambda = -\Gamma_\lambda \nabla G_\lambda \tag{5.65}$$

The radiation source term in the energy equation is given as

$$-\nabla q_r = \sum_{\text{All Bands}} -\nabla q_{r,\lambda} = \sum_{\text{All Bands}} a_\lambda (G_\lambda - n^2 G_{b\lambda}) \tag{5.66}$$

5.3.3.2. Anisotropic Scattering

Included in the P-1 radiation model is the capability for modeling anisotropic scattering. Ansys Fluent models anisotropic scattering by means of a linear-anisotropic scattering phase function:

$$\Phi\left(\vec{s}' \cdot \vec{s}\right) = 1 + C\vec{s}' \cdot \vec{s} \tag{5.67}$$

where

$\vec{s}$ = unit vector in the direction of scattering

$\vec{s}'$ = unit vector in the direction of the incident radiation

C = linear-anisotropic phase function coefficient, which is a property of the fluid

Value of C ranges from –1 to 1. A positive value indicates that more radiant energy is scattered forward than backward, and a negative value means that more radiant energy is scattered backward than forward. A zero value defines isotropic scattering (that is, scattering that is equally likely in all

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

directions), which is the default in Ansys Fluent. You should modify the default value only if you are certain of the anisotropic scattering behavior of the material in your problem.

5.3.3.3. Particulate Effects in the P-1 Model

When your Ansys Fluent model includes a dispersed second phase of particles, you can include the effect of particles in the P-1 radiation model. Note that when particles are present, Ansys Fluent ignores scattering in the gas phase. That is, Equation 5.68 (p. 192) assumes that all scattering is due to particles.

For a gray, absorbing, emitting, and scattering medium containing absorbing, emitting, and scattering particles, the transport equation for the incident radiation can be written as

$$\nabla\cdot\left(\Gamma\nabla G\right)+4\pi\left(an^2\frac{\sigma T^4}{\pi}+E_p\right)-\left(a+a_p\right)G=0 \tag{5.68}$$

where E_p is the equivalent emission of the particles, a_p is the equivalent absorption coefficient, and n is the refractive index of the medium. These are defined as follows:

$$E_p=\lim_{V\to 0}\sum_{n=1}^{N}\varepsilon_{pn}A_{pn}\frac{\sigma T_{pn}^4}{\pi V} \tag{5.69}$$

and

$$a_p=\lim_{V\to 0}\sum_{n=1}^{N}\varepsilon_{pn}\frac{A_{pn}}{V} \tag{5.70}$$

In Equation 5.69 (p. 192) and Equation 5.70 (p. 192), ε_{pn}, A_{pn}, and T_{pn} are the emissivity, projected area, and temperature of particle n. The summation is over N particles in volume V. These quantities are computed during particle tracking in Ansys Fluent.

The projected area A_{pn} of particle n is defined as

$$A_{pn}=\frac{\pi d_{pn}^2}{4} \tag{5.71}$$

where d_{pn} is the diameter of the n^{th} particle.

The quantity Γ in Equation 5.68 (p. 192) is defined as

$$\Gamma=\frac{1}{3\left(a+a_p+\sigma_p\right)} \tag{5.72}$$

where the equivalent particle scattering factor is computed during particle tracking and is defined as

$$\sigma_p=\lim_{V\to 0}\sum_{n=1}^{N}\left(1-f_{pn}\right)\left(1-\varepsilon_{pn}\right)\frac{A_{pn}}{V} \tag{5.73}$$

Here, f_{pn} is the scattering factor associated with the n^{th} particle.

Heat sources (sinks) due to particle radiation are included in the energy equation as follows:

$$-\nabla\cdot q_r=-4\pi\left(an^2\frac{\sigma T^4}{\pi}+E_p\right)+\left(a+a_p\right)G \tag{5.74}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

5.3.3.4. Boundary Condition Treatment for the P-1 Model at Walls

To get the boundary condition for the incident radiation equation when modeling gray radiation, the dot product of the outward normal vector $\vec{n}$ and Equation 5.59 (p. 190) is computed:

$$q_r \cdot \vec{n} = -\Gamma \nabla G \cdot \vec{n} \tag{5.75}$$

$$q_{r,w} = -\Gamma \frac{\partial G}{\partial n} \tag{5.76}$$

Thus the flux of the incident radiation, G, at a wall is $-q_{r,w}$. The wall radiative heat flux is computed using the following boundary condition:

$$I_w(\vec{r},\vec{s}) = f_w(\vec{r},\vec{s}) \tag{5.77}$$

$$f_w(\vec{r},\vec{s}) = \epsilon_w \frac{n^2 \sigma T_w^4}{\pi} + \rho_w I(\vec{r},-\vec{s}) \tag{5.78}$$

where ρ_w is the wall reflectivity. The Marshak boundary condition is then used to eliminate the angular dependence [500] (p. 1082):

$$\int_0^{2\pi} I_w(\vec{r},\vec{s})\vec{n}\cdot\vec{s}\,d\Omega = \int_0^{2\pi} f_w(\vec{r},\vec{s})\vec{n}\cdot\vec{s}\,d\Omega \tag{5.79}$$

Substituting Equation 5.77 (p. 193) and Equation 5.78 (p. 193) into Equation 5.79 (p. 193) and performing the integrations yields

$$q_{r,w} = -\frac{4\pi\varepsilon_w \frac{n^2\sigma T_w^4}{\pi} - \left(1-\rho_w\right)G_w}{2\left(1+\rho_w\right)} \tag{5.80}$$

If it is assumed that the walls are diffuse gray surfaces, then $\rho_w = 1-\varepsilon_w$, and Equation 5.80 (p. 193) becomes

$$q_{r,w} = -\frac{\varepsilon_w}{2(2-\varepsilon_w)}\left(4n^2\sigma T_w^4 - G_w\right) \tag{5.81}$$

Equation 5.81 (p. 193) is used to compute $q_{r,w}$ for the energy equation and for the incident radiation equation boundary conditions.

When using the gray-band model, the flux at the wall can be written as the following (assuming that the wall is a diffuse surface):

$$q_{\lambda,w} = -\frac{\varepsilon_{\lambda,w}}{2(2-\varepsilon_{\lambda,w})}\left(4n^2[F(0\to n\lambda_2 T) - F(0\to n\lambda_1 T)]\sigma T^4 - G_{\lambda,w}\right) \tag{5.82}$$

where $\varepsilon_{\lambda,w}$ is the wall emissivity, $q_{\lambda,w}$ is the spectral radiative flux at the wall, and $G_{\lambda,w}$ is the spectral incident radiation at the wall.

5.3.3.5. Boundary Condition Treatment for the P-1 Model at Flow Inlets and Exits

The net radiative heat flux at flow inlets and outlets is computed in the same manner as at walls, as described above. Ansys Fluent assumes that the emissivity of all flow inlets and outlets is 1.0 (black body absorption) unless you choose to redefine this boundary treatment.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Ansys Fluent includes an option that allows you to use different temperatures for radiation and convection at inlets and outlets. This can be useful when the temperature outside the inlet or outlet differs considerably from the temperature in the enclosure. For details, see Defining Boundary Conditions for Radiation in the User's Guide.

5.3.4. Rosseland Radiation Model Theory

The Rosseland or diffusion approximation for radiation is valid when the medium is optically thick ($\left(a+\sigma_s\right)L \gg 1$), and is recommended for use in problems where the optical thickness is greater than 3. It can be derived from the P-1 model equations, with some approximations. This section provides details about the equations used in the Rosseland model. For information about setting up the model, see Using the Radiation Models in the User's Guide.

5.3.4.1. The Rosseland Model Equations

As with the P-1 model, the radiative heat flux vector in a gray medium can be approximated by Equation 5.59 (p. 190):

$$q_r = -\Gamma \nabla G \tag{5.83}$$

where Γ is given by Equation 5.58 (p. 190).

The Rosseland radiation model differs from the P-1 model in that the Rosseland model assumes that the intensity is the black-body intensity at the gas temperature. (The P-1 model actually calculates a transport equation for G.) Thus $G = 4\sigma n^2 T^4$, where n is the refractive index. Substituting this value for G into Equation 5.83 (p. 194) yields

$$q_r = -16\sigma\Gamma n^2 T^3 \nabla T \tag{5.84}$$

Since the radiative heat flux has the same form as the Fourier conduction law, it is possible to write

$$q = q_c + q_r \tag{5.85}$$

$$q = -\left(k + k_r\right)\nabla T \tag{5.86}$$

$$k_r = 16\sigma\Gamma n^2 T^3 \tag{5.87}$$

where k is the thermal conductivity and k_r is the radiative conductivity. Equation 5.85 (p. 194) is used in the energy equation to compute the temperature field.

5.3.4.2. Anisotropic Scattering

The Rosseland model allows for anisotropic scattering, using the same phase function (Equation 5.67 (p. 191)) described for the P-1 model in Anisotropic Scattering (p. 191).

5.3.4.3. Boundary Condition Treatment at Walls

Since the diffusion approximation is not valid near walls, it is necessary to use a temperature slip boundary condition. The radiative heat flux at the wall boundary, $q_{r,w}$, is defined using the slip coefficient ψ:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$q_{r,w} = -\frac{\sigma\left(T_w^4 - T_g^4\right)}{\psi} \tag{5.88}$$

where T_w is the wall temperature, T_g is the temperature of the gas at the wall, and the slip coefficient ψ is approximated by a curve fit to the plot given in [593] (p. 1087):

$$\psi = \begin{cases} 1/2 & N_w < 0.01 \\ \frac{2x^3+3x^2-12x+7}{54} & 0.01 \le N_w \le 10 \\ 0 & N_w > 10 \end{cases} \tag{5.89}$$

where N_w is the conduction to radiation parameter at the wall:

$$N_w = \frac{k\left(a+\sigma_s\right)}{4\sigma T_w^3} \tag{5.90}$$

and $x = \log_{10} N_w$.

5.3.4.4. Boundary Condition Treatment at Flow Inlets and Exits

No special treatment is required at flow inlets and outlets for the Rosseland model. The radiative heat flux at these boundaries can be determined using Equation 5.85 (p. 194).

5.3.5. Discrete Transfer Radiation Model (DTRM) Theory

The main assumption of the DTRM is that the radiation leaving the surface element in a certain range of solid angles can be approximated by a single ray. This section provides details about the equations used in the DTRM. For information about setting up the model, see Setting Up the DTRM in the User's Guide.

5.3.5.1. The DTRM Equations

The equation for the change of radiant intensity, dI, along a path, ds, can be written as

$$\frac{dI}{ds} + aI = \frac{a\sigma T^4}{\pi} \tag{5.91}$$

where

a = gas absorption coefficient

I = intensity

T = gas local temperature

σ = Stefan-Boltzmann constant ($5.669 \times 10^{-8} W/m^2 - K^4$)

Here, the refractive index is assumed to be unity. The DTRM integrates Equation 5.91 (p. 195) along a series of rays emanating from boundary faces. If a is constant along the ray, then $I(s)$ can be estimated as

$$I(s) = \frac{\sigma T^4}{\pi}\left(1 - e^{-as}\right) + I_0 e^{-as} \tag{5.92}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

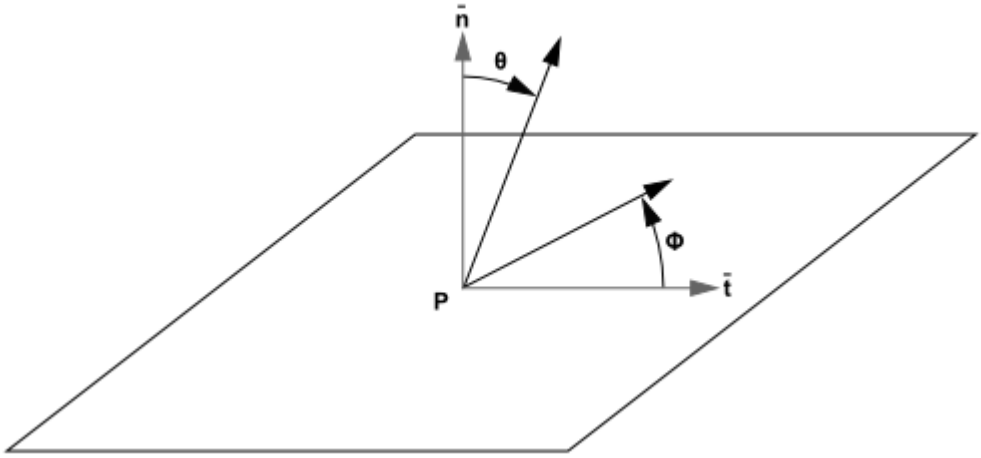
where I_0 is the radiant intensity at the start of the incremental path, which is determined by the appropriate boundary condition (see the description of boundary conditions, below). The energy source in the fluid due to radiation is then computed by summing the change in intensity along the path of each ray that is traced through the fluid control volume.

The "ray tracing" technique used in the DTRM can provide a prediction of radiative heat transfer between surfaces without explicit view factor calculations. The accuracy of the model is limited mainly by the number of rays traced and the computational mesh.

5.3.5.2. Ray Tracing

The ray paths are calculated and stored prior to the fluid flow calculation. At each radiating face, rays are fired at discrete values of the polar and azimuthal angles (see Figure 5.2: Angles θ and ϕ Defining the Hemispherical Solid Angle About a Point P (p. 196)). To cover the radiating hemisphere, θ is varied from 0 to $\frac{\pi}{2}$ and φ from 0 to 2π. Each ray is then traced to determine the control volumes it intercepts as well as its length within each control volume. This information is then stored in the radiation file, which must be read in before the fluid flow calculations begin.

Figure 5.2: Angles θ and ϕ Defining the Hemispherical Solid Angle About a Point P



5.3.5.3. Clustering

DTRM is computationally very expensive when there are too many surfaces to trace rays from and too many volumes crossed by the rays. To reduce the computational time, the number of radiating surfaces and absorbing cells is reduced by clustering surfaces and cells into surface and volume "clusters". The volume clusters are formed by starting from a cell and simply adding its neighbors and their neighbors until a specified number of cells per volume cluster is collected. Similarly, surface clusters are made by starting from a face and adding its neighbors and their neighbors until a specified number of faces per surface cluster is collected.

The incident radiation flux, q_{in}, and the volume sources are calculated for the surface and volume clusters respectively. These values are then distributed to the faces and cells in the clusters to calculate the wall and cell temperatures. Since the radiation source terms are highly nonlinear (proportional to the fourth power of temperature), care must be taken to calculate the average temperatures of surface and volume clusters and distribute the flux and source terms appropriately among the faces and cells forming the clusters.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

The surface and volume cluster temperatures are obtained by area and volume averaging as shown in the following equations:

$$T_{sc}=\left(\frac{\sum_{f} A_f T_f^4}{\sum A_f}\right)^{1/4} \tag{5.93}$$

$$T_{vc}=\left(\frac{\sum_{c} V_c T_c^4}{\sum V_c}\right)^{1/4} \tag{5.94}$$

where T_{sc} and T_{vc} are the temperatures of the surface and volume clusters respectively, A_f and T_f are the area and temperature of face f, and V_c and T_c are the volume and temperature of cell c. The summations are carried over all faces of a surface cluster and all cells of a volume cluster.

5.3.5.4. Boundary Condition Treatment for the DTRM at Walls

The radiation intensity approaching a point on a wall surface is integrated to yield the incident radiative heat flux, q_{in}, as

$$q_{in}=\int_{\vec{s}\cdot\vec{n}>0} I_{in}\vec{s}\cdot\vec{n}d\Omega \tag{5.95}$$

where Ω is the hemispherical solid angle, I_{in} is the intensity of the incoming ray, $\vec{s}$ is the ray direction vector, and $\vec{n}$ is the normal pointing out of the domain. The net radiative heat flux from the surface, q_{out}, is then computed as a sum of the reflected portion of q_{in} and the emissive power of the surface:

$$q_{out}=\left(1-\varepsilon_w\right)q_{in}+\varepsilon_w\sigma T_w^4 \tag{5.96}$$

where T_w is the surface temperature of the point P on the surface and ε_w is the wall emissivity which you enter as a boundary condition. Ansys Fluent incorporates the radiative heat flux (Equation 5.96 (p. 197)) in the prediction of the wall surface temperature. Equation 5.96 (p. 197) also provides the surface boundary condition for the radiation intensity I_0 of a ray emanating from the point P, as

$$I_0=\frac{q_{out}}{\pi} \tag{5.97}$$

5.3.5.5. Boundary Condition Treatment for the DTRM at Flow Inlets and outlets

The net radiative heat flux at flow inlets and outlets is computed in the same manner as at walls, as described above. Ansys Fluent assumes that the emissivity of all flow inlets and outlets is 1.0 (black body absorption) unless you choose to redefine this boundary treatment.

Ansys Fluent includes an option that allows you to use different temperatures for radiation and convection at inlets and outlets. This can be useful when the temperature outside the inlet or outlet differs considerably from the temperature in the enclosure. For details, see Defining Boundary Conditions for Radiation in the User's Guide.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

5.3.6. Discrete Ordinates (DO) Radiation Model Theory

The discrete ordinates (DO) radiation model solves the radiative transfer equation (RTE) for a finite number of discrete solid angles, each associated with a vector direction $\vec{s}$ fixed in the global Cartesian system (x,y,z). You control the fineness of the angular discretization, analogous to choosing the number of rays for the DTRM. Unlike the DTRM, however, the DO model does not perform ray tracing. Instead, the DO model transforms Equation 5.56 (p. 189) into a transport equation for radiation intensity in the spatial coordinates (x,y,z). The DO model solves for as many transport equations as there are directions $\vec{s}$. The solution method is identical to that used for the fluid flow and energy equations.

Two implementations of the DO model are available in Ansys Fluent: uncoupled and (energy) coupled. The uncoupled implementation is sequential in nature and uses a conservative variant of the DO model called the finite-volume scheme [110] (p. 1059), [535] (p. 1084), and its extension to unstructured meshes [463] (p. 1080). In the uncoupled case, the equations for the energy and radiation intensities are solved one by one, assuming prevailing values for other variables.

Alternatively, in the coupled ordinates method (or COMET) [412] (p. 1077), the discrete energy and intensity equations at each cell are solved simultaneously, assuming that spatial neighbors are known. The advantages of using the coupled approach is that it speeds up applications involving high optical thicknesses and/or high scattering coefficients. Such applications slow down convergence drastically when the sequential approach is used. For information about setting up the model, see Setting Up the DO Model in the User's Guide.

5.3.6.1. The DO Model Equations

The DO model considers the radiative transfer equation (RTE) in the direction $\vec{s}$ as a field equation. Thus, Equation 5.56 (p. 189) is written as

$$\begin{aligned}\nabla\cdot\left(I\left(\vec{r},\vec{s}\right)\vec{s}\right)+\left(a+\sigma_s\right)I\left(\vec{r},\vec{s}\right) &= an^2\frac{\sigma T^4}{\pi}\\ &\quad+\frac{\sigma_s}{4\pi}\int_0^{4\pi} I\left(\vec{r},\vec{s}'\right)\Phi\left(\vec{s}\cdot\vec{s}'\right)d\Omega'\end{aligned} \tag{5.98}$$

Ansys Fluent also allows the modeling of non-gray radiation using a gray-band model. The RTE for the spectral intensity $I_\lambda\left(\vec{r},\vec{s}\right)$ can be written as [[445] (p. 1078) equation 9.22]

$$\begin{aligned}\nabla\cdot\left(I_\lambda\left(\vec{r},\vec{s}\right)\vec{s}\right)+\left(a_\lambda+\sigma_s\right)I_\lambda\left(\vec{r},\vec{s}\right) &= a_\lambda I_{b\lambda}\\ &\quad+\frac{\sigma_s}{4\pi}\int_0^{4\pi} I_\lambda\left(\vec{r},\vec{s}'\right)\Phi\left(\vec{s}\cdot\vec{s}'\right)d\Omega'\end{aligned} \tag{5.99}$$

Here λ is the wavelength, a_λ is the spectral absorption coefficient, and $I_{b\lambda}$ is the black body intensity given by the Planck function. The scattering coefficient, the scattering phase function, and the refractive index n are assumed independent of wavelength.

The non-gray DO implementation divides the radiation spectrum into N wavelength bands, which need not be contiguous or equal in extent. The wavelength intervals are supplied by you, and correspond to values in vacuum (n=1). The RTE is integrated over each wavelength interval, resulting in transport equations for the quantity $I_\lambda\Delta\lambda$, the radiant energy contained in the wavelength band $\Delta\lambda$. The behavior in each band is assumed gray. The black body emission in the wavelength band per unit solid angle is written as

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$\left[F\left(0\rightarrow n\lambda_2 T\right)-F\left(0\rightarrow n\lambda_1 T\right)\right]n^2\frac{\sigma T^4}{\pi} \tag{5.100}$$

where $F\left(0\rightarrow n\lambda T\right)$ is the fraction of radiant energy emitted by a black body [444] (p. 1078) in the wavelength interval from 0 to λ at temperature T in a medium of refractive index n. λ_2 and λ_1 are the wavelength boundaries of the band.

The total intensity $I\left(\vec{r},\vec{s}\right)$ in each direction $\vec{s}$ at position $\vec{r}$ is computed using

$$I\left(\vec{r},\vec{s}\right)=\sum_k I_{\lambda_k}\left(\vec{r},\vec{s}\right)\Delta\lambda_k \tag{5.101}$$

where the summation is over the wavelength bands.

Boundary conditions for the non-gray DO model are applied on a band basis. The treatment within a band is the same as that for the gray DO model.

5.3.6.2. Energy Coupling and the DO Model

The coupling between energy and radiation intensities at a cell (which is also known as COMET) [412] (p. 1077) accelerates the convergence of the finite volume scheme for radiative heat transfer. This method results in significant improvement in the convergence for applications involving optical thicknesses greater than 10. This is typically encountered in glass-melting applications. This feature is advantageous when scattering is significant, resulting in strong coupling between directional radiation intensities. This DO model implementation is utilized in Ansys Fluent by enabling the **DO/Energy Coupling** option for the DO model in the **Radiation Model** dialog box. The discrete energy equations for the coupled method are presented below.

The energy equation when integrated over a control volume i, yields the discrete energy equation:

$$\sum_{j=1}^{N}\mu_{ij}^T T_j-\beta_i^T T_i=\alpha_i^T\sum_{k=1}^{L}I_i^k\omega_k-S_i^T+S_i^h \tag{5.102}$$

where

$$\alpha_i^T = \kappa\Delta V_i$$

$$\beta_i^T = 16\kappa\sigma T_i^{*3}\Delta V_i$$

$$S_i^T = 12\kappa\sigma T_i^{*4}\Delta V_i$$

κ = absorption coefficient

ΔV control volume

The coefficient μ_{ij}^T and the source term S_i^h are due to the discretization of the convection and diffusion terms as well as the non-radiative source terms.

Combining the discretized form of Equation 5.98 (p. 198) and the discretized energy equation, Equation 5.102 (p. 199), yields [412] (p. 1077):

$$\vec{P}_i\vec{q}_i+\vec{r}_i=0 \tag{5.103}$$

where

$$\vec{q}_i = \begin{bmatrix} I_i^1 \\ I_i^2 \\ \vdots \\ \vdots \\ I_i^L \\ T_i \end{bmatrix} \tag{5.104}$$

$$\vec{P}_i = \begin{bmatrix} M_{ii}^1 + \eta_i^{11} + \gamma_i^1 \omega_1 & \eta_i^{12} + \gamma_i^1 \omega_2 & \ldots \beta_i^1 \\ \eta_i^{21} + \gamma_i^2 \omega_1 & M_{ii}^2 + \eta_i^{22} + \gamma_i^2 \omega_2 & \ldots \beta_i^2 \\ \vdots & & \\ -\alpha_i^T \omega_1 & -\alpha_i^T \omega_2 & \ldots M_{ii}^T \end{bmatrix} \tag{5.105}$$

$$\vec{r}_i = \begin{bmatrix} \sum_{j=1, i \neq j}^{N} \mu_{ij}^l I_j^l - S_i^1 - S_i^B \\ \vdots \\ \sum_{j=1, i \neq j}^{N} \mu_{ij}^T T_j + S_i^T + S_i^h \end{bmatrix} \tag{5.106}$$

5.3.6.2.1. Limitations of DO/Energy Coupling

There are some instances when using DO/Energy coupling is not recommended or is incompatible with certain models:

- DO/Energy coupling is not recommended for cases with weak coupling between energy and directional radiation intensities. This may result in slower convergence of the coupled approach compared to the sequential approach.
- DO/Energy coupling is not available when solving enthalpy equations instead of temperature equations. Specifically, DO/Energy coupling is not compatible with the non-premixed or partially premixed combustion models.

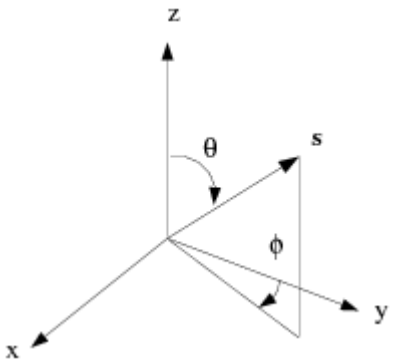
To find out how to apply DO/Energy coupling, refer to Setting Up the DO Model in the User's Guide.

5.3.6.3. Angular Discretization and Pixelation

Each octant of the angular space 4π at any spatial location is discretized into $N_\theta \times N_\varphi$ solid angles of extent ω_i, called control angles. The angles θ and φ are the polar and azimuthal angles respectively, and are measured with respect to the global Cartesian system (x, y, z) as shown in Figure 5.3: Angular Coordinate System (p. 201). The θ and φ extents of the control angle, $\Delta\theta$ and $\Delta\varphi$, are constant. In two-dimensional calculations, only four octants are solved due to symmetry, making a total of $4N_\theta N_\varphi$ directions in all. In three-dimensional calculations, a total of $8N_\theta N_\varphi$ directions are solved. In the case of the non-gray model, $4N_\theta N_\varphi$ or $8N_\theta N_\varphi$ equations are solved for each band.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Figure 5.3: Angular Coordinate System



When Cartesian meshes are used, it is possible to align the global angular discretization with the control volume face, as shown in Figure 5.4: Face with No Control Angle Overhang (p. 201). For generalized unstructured meshes, however, control volume faces do not in general align with the global angular discretization, as shown in Figure 5.5: Face with Control Angle Overhang (p. 202), leading to the problem of control angle overhang [463] (p. 1080).

Figure 5.4: Face with No Control Angle Overhang

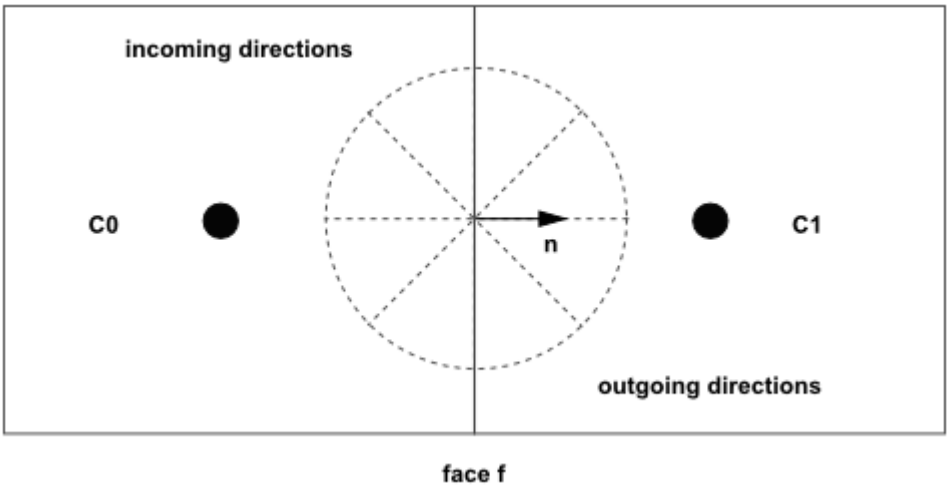
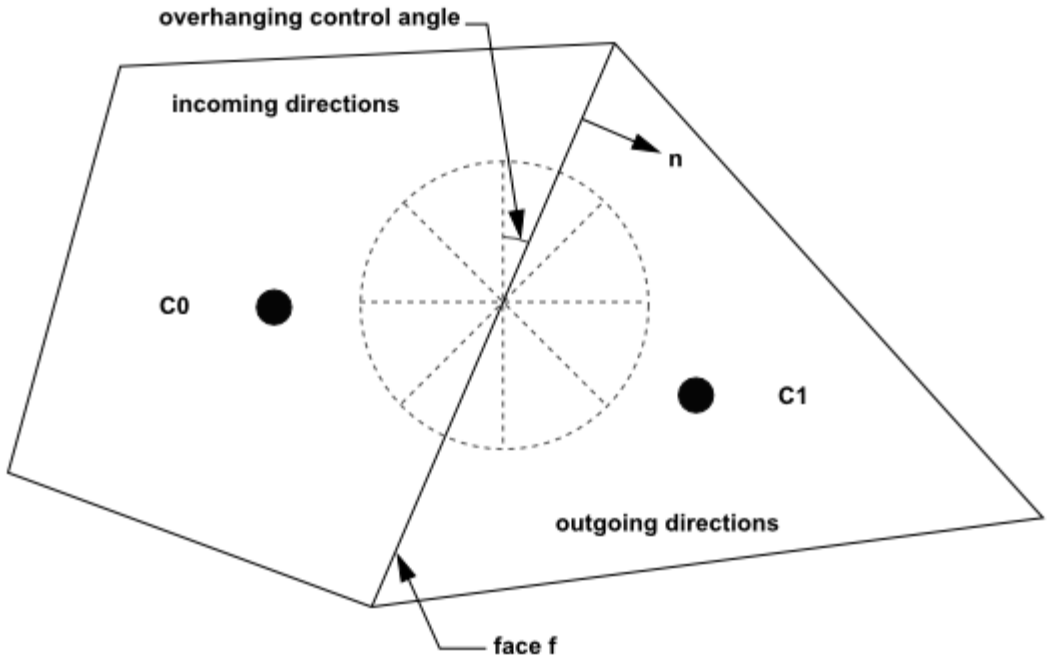
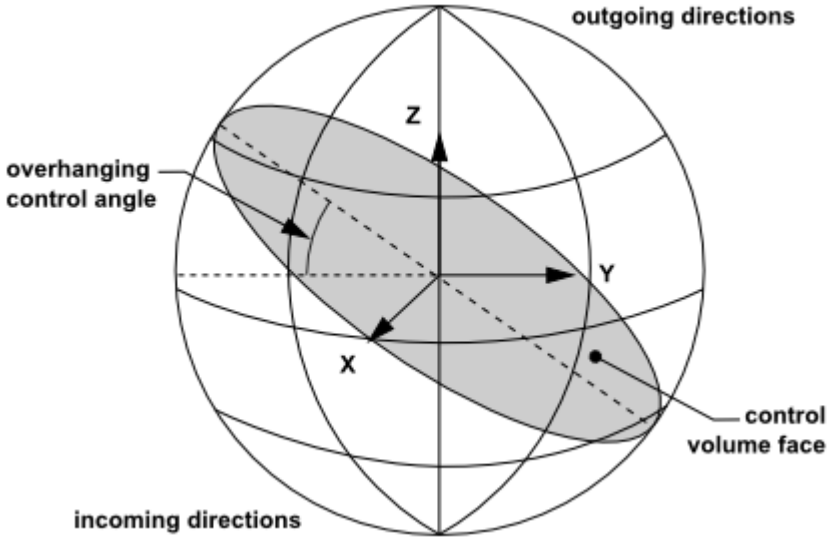


Figure 5.5: Face with Control Angle Overhang



Essentially, control angles can straddle the control volume faces, so that they are partially incoming and partially outgoing to the face. Figure 5.6: Face with Control Angle Overhang (3D) (p. 202) shows a 3D example of a face with control angle overhang.

Figure 5.6: Face with Control Angle Overhang (3D)

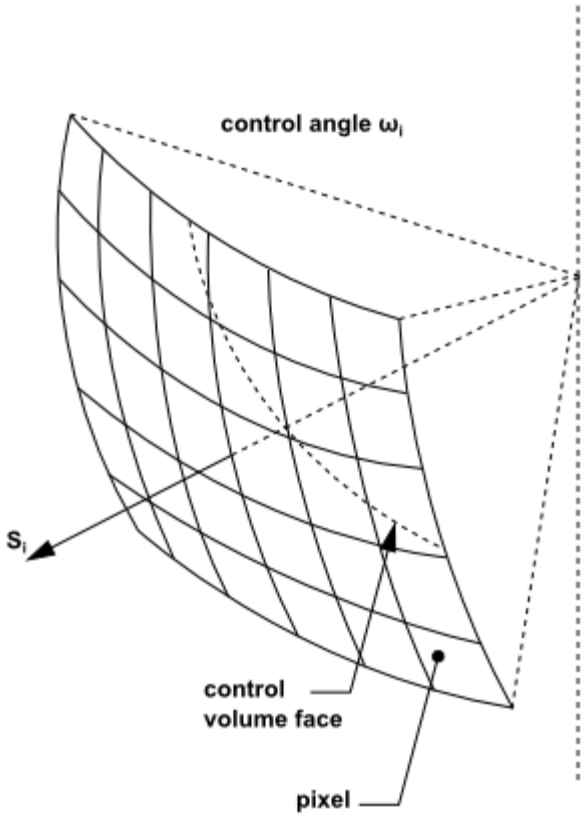


The control volume face cuts the sphere representing the angular space at an arbitrary angle. The line of intersection is a great circle. Control angle overhang may also occur as a result of reflection and refraction. It is important in these cases to correctly account for the overhanging fraction. This is done through the use of pixelation [463] (p. 1080).

Each overhanging control angle is divided into $N_{\theta_p} \times N_{\varphi_p}$ pixels, as shown in Figure 5.7: Pixelation of Control Angle (p. 203).

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Figure 5.7: Pixelation of Control Angle



The energy contained in each pixel is then treated as incoming or outgoing to the face. The influence of overhang can therefore be accounted for within the pixel resolution. Ansys Fluent allows you to choose the pixel resolution. For problems involving gray-diffuse radiation, the default pixelation of 1×1 is usually sufficient. For problems involving symmetry, periodic, specular, or semi-transparent boundaries, a pixelation of 3×3 is recommended. You should be aware, however, that increasing the pixelation adds to the cost of computation.

5.3.6.4. Anisotropic Scattering

The DO implementation in Ansys Fluent admits a variety of scattering phase functions. You can choose an isotropic phase function, a linear anisotropic phase function, a Delta-Eddington phase function, or a user-defined phase function. The linear anisotropic phase function is described in Equation 5.67 (p. 191). The Delta-Eddington function takes the following form:

$$\Phi\left(\vec{s}\cdot\vec{s}'\right)=2f\delta\left(1-\vec{s}\cdot\vec{s}'\right)+\left(1-f\right)\left(1+C\vec{s}\cdot\vec{s}'\right) \tag{5.107}$$

Here, f is the forward-scattering factor and $\delta\left(\vec{s}\cdot\vec{s}'\right)$ is the Dirac delta function. The f term essentially cancels a fraction f of the out-scattering; therefore, for $f=1$, the Delta-Eddington phase function will cause the intensity to behave as if there is no scattering at all. C is the asymmetry factor. When the Delta-Eddington phase function is used, you will specify values for f and C.

When a user-defined function is used to specify the scattering phase function, Ansys Fluent assumes the phase function to be of the form

$$\Phi\left(\vec{s}\cdot\vec{s}'\right)=2f\delta\left(1-\vec{s}\cdot\vec{s}'\right)+\left(1-f\right)\Phi^{*}\left(\vec{s}\cdot\vec{s}'\right) \tag{5.108}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

The user-defined function will specify Φ^* and the forward-scattering factor f.

The scattering phase functions available for gray radiation can also be used for non-gray radiation. However, the scattered energy is restricted to stay within the band.

5.3.6.5. Particulate Effects in the DO Model

The DO model allows you to include the effect of a discrete second phase of particulates on radiation. In this case, Ansys Fluent will neglect all other sources of scattering in the gas phase.

The contribution of the particulate phase appears in the RTE as:

$$\nabla \cdot \left(I\vec{s}\right) + \left(a + a_p + \sigma_p\right) I\left(\vec{r},\vec{s}\right) = an^2\frac{\sigma T^4}{\pi} + E_p + \frac{\sigma_p}{4\pi}\int_0^{4\pi} I\left(\vec{r},\vec{s}'\right)\Phi\left(\vec{s}\cdot\vec{s}'\right)d\Omega' \tag{5.109}$$

where a_p is the equivalent absorption coefficient due to the presence of particulates, and is given by Equation 5.70 (p. 192). The equivalent emission E_p is given by Equation 5.69 (p. 192). The equivalent particle scattering factor σ_p, defined in Equation 5.73 (p. 192), is used in the scattering terms.

For non-gray radiation, absorption, emission, and scattering due to the particulate phase are included in each wavelength band for the radiation calculation. Particulate emission and absorption terms are also included in the energy equation.

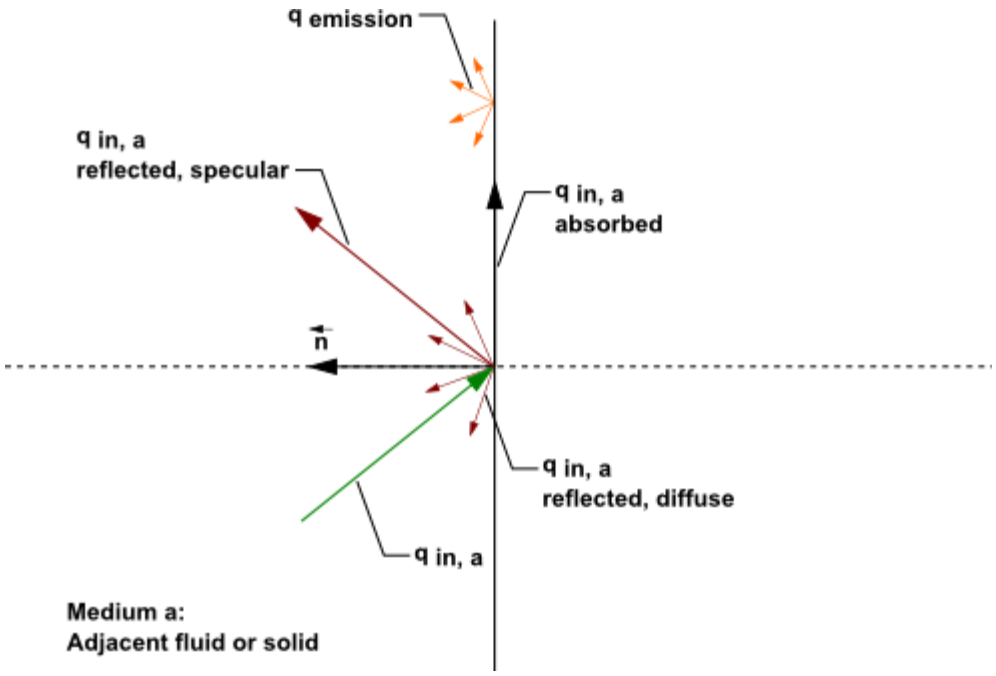
5.3.6.6. Boundary and Cell Zone Condition Treatment at Opaque Walls

The discrete ordinates radiation model allows the specification of opaque walls that are interior to a domain (with adjacent fluid or solid zones on both sides of the wall), or external to the domain (with an adjacent fluid or solid zone on one side, only). Opaque walls are treated as gray if gray radiation is being computed, or non-gray if the non-gray DO model is being used.

Figure 5.8: DO Radiation on Opaque Wall (p. 205) shows a schematic of radiation on an opaque wall in Ansys Fluent.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Figure 5.8: DO Radiation on Opaque Wall



The diagram in Figure 5.8: DO Radiation on Opaque Wall (p. 205) shows incident radiation $q_{in,a}$ on side a of an opaque wall. Some of the radiant energy is reflected diffusely and specularly, depending on the diffuse fraction f_d for side a of the wall that you specify as a boundary condition.

Some of the incident radiation is absorbed at the surface of the wall and some radiation is emitted from the wall surface as shown in Figure 5.8: DO Radiation on Opaque Wall (p. 205). The amount of incident radiation absorbed at the wall surface and the amount emitted back depends on the emissivity of that surface and the diffuse fraction. For non-gray DO models, you must specify internal emissivity for each wavelength band. Radiation is not transmitted through an opaque wall.

Radiant incident energy that impacts an opaque wall can be reflected back to the surrounding medium and absorbed by the wall. The radiation that is reflected can be diffusely reflected and/or specularly reflected, depending on the diffuse fraction f_d. If q_{in} is the amount of radiative energy incident on the opaque wall, then the following general quantities are computed by Ansys Fluent for opaque walls:

- emission from the wall surface $= n^2 \varepsilon_w \sigma T_w^4$
- diffusely reflected energy $= f_d \left(1 - \varepsilon_w\right) q_{in}$
- specularly reflected energy $= \left(1 - f_d\right)\left(1 - \varepsilon_w\right) q_{in}$
- absorption at the wall surface $= \varepsilon_w q_{in}$

where f_d is the diffuse fraction, n is the refractive index of the adjacent medium, ε_w is the wall emissivity, σ is the Stefan-Boltzmann Constant, and T_w is the wall temperature.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Note that although Ansys Fluent uses emissivity in its computation of radiation quantities, it is not available for postprocessing. Absorption at the wall surface assumes that the absorptivity is equal to the emissivity. For a purely diffused wall, f_d is equal to 1 and there is no specularly reflected energy. Similarly, for a purely specular wall, f_d is equal to 0 and there is no diffusely reflected energy. A diffuse fraction between 0 and 1 will result in partially diffuse and partially reflected energy.

5.3.6.6.1. Gray Diffuse Walls

For gray diffuse radiation, the incident radiative heat flux, q_{in}, at the wall is

The net radiative flux leaving the surface is given by

$$q_{in} = -\int_{4\pi} I_{in}\vec{s}\cdot\vec{n}d\Omega, \vec{s}\cdot\vec{n}<0 \tag{5.110}$$

$$q_{out} = \left(1-\varepsilon_w\right)q_{in} + n^2\varepsilon_w\sigma T_w^4 \tag{5.111}$$

where

n = the refractive index of the medium next to the wall

ε_w = wall emissivity

σ = Stefan-Boltzmann Constant

T_w = wall temperature

This equation is also valid for specular radiation with emissivity = 0.

The boundary intensity for all outgoing directions $\vec{s}$ at the wall is given by

$$I_0 = \frac{q_{out}}{\pi} \tag{5.112}$$

5.3.6.6.2. Non-Gray Diffuse Walls

There is a special set of equations that apply uniquely to non-gray diffuse opaque walls. These equations assume that the absorptivity is equal to the emissivity for the wall surface. For non-gray diffuse radiation, the incident radiative heat flux $q_{in,\lambda}$ in the band $\Delta\lambda$ at the wall is

$$q_{in,\lambda} = \Delta\lambda \int_{\vec{s}\cdot\vec{n}>0} I_{in,\lambda}\vec{s}\cdot\vec{n}d\Omega \tag{5.113}$$

The net radiative flux leaving the surface in the band $\Delta\lambda$ is given by

$$q_{out,\lambda} = \left(1-\varepsilon_{w\lambda}\right)q_{in,\lambda} + \varepsilon_{w\lambda}\left[F\left(0\rightarrow n\lambda_2 T_w\right) - F\left(0\rightarrow n\lambda_1 T_w\right)\right]n^2\sigma T_w^4 \tag{5.114}$$

where $\varepsilon_{w\lambda}$ is the wall emissivity in the band. $F\left(n,\lambda,T\right)$ provides the Planck distribution function. This defines the emittance for each radiation band as a function of the temperature of the source surface. The boundary intensity for all outgoing directions $\vec{s}$ in the band $\Delta\lambda$ at the wall is given by

$$I_{0\lambda} = \frac{q_{out,\lambda}}{\pi\Delta\lambda} \tag{5.115}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

5.3.6.7. Cell Zone and Boundary Condition Treatment at Semi-Transparent Walls

Ansys Fluent allows the specification of interior and exterior semi-transparent walls for the DO model. In the case of interior semi-transparent walls, incident radiation can pass through the wall and be transmitted to the adjacent medium (and possibly refracted), it can be reflected back into the surrounding medium, and absorbed through the wall thickness. Transmission and reflection can be diffuse and/or specular. You specify the diffuse fraction for all transmitted and reflected radiation; the rest is treated specularly. For exterior semi-transparent walls, there are two possible sources of radiation on the boundary wall: an irradiation beam from outside the computational domain and incident radiation from cells in adjacent fluid or solid zones.

For non-gray radiation, semi-transparent wall boundary conditions are applied on a per-band basis. The radiant energy within a band is transmitted, reflected, and refracted as in the gray case; there is no transmission, reflection, or refraction of radiant energy from one band to another.

By default the DO equations are solved in all fluid zones, but not in any solid zones. Therefore, if you have an adjacent solid zone for your thin wall, you will need to specify the solid zone as participating in radiation in the **Solid** dialog box as part of the boundary condition setup.

Important:

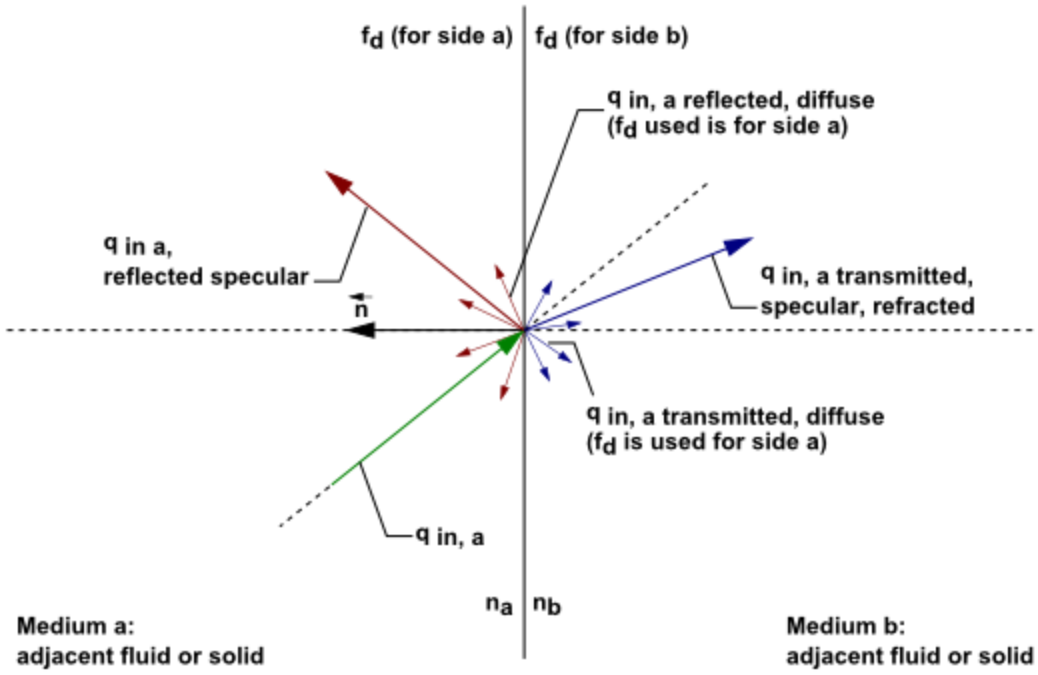
If you are interested in the detailed temperature distribution inside your semi-transparent media, then you will need to model a semi-transparent wall as a solid zone with adjacent fluid zone(s), and treat the solid as a semi-transparent medium. This is discussed in a subsequent section.

5.3.6.7.1. Semi-Transparent Interior Walls

Figure 5.9: DO Radiation on Interior Semi-Transparent Wall (p. 208) shows a schematic of an interior (two-sided) wall that is treated as semi-transparent in Ansys Fluent and has zero-thickness. Incident radiant energy depicted by $q_{in,a}$ can pass through the semi-transparent wall if and only if the contiguous fluid or solid cell zones participate in radiation, thereby allowing the radiation to be coupled. Radiation coupling is set when a wall is specified as semi-transparent. Note that by default, radiation is *not* coupled and you will need to explicitly specify radiation coupling on the interior wall by changing the boundary condition type to semi-transparent in the **Wall** dialog box (under the **Radiation** tab).

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Figure 5.9: DO Radiation on Interior Semi-Transparent Wall



Incident radiant energy that is transmitted through a semi-transparent wall can be transmitted specularly and diffusely. Radiation can also be reflected at the interior wall back to the surrounding medium if the refractive index n_a for the fluid zone that represents medium a is different than the refractive index n_b for medium b. Reflected radiation can be reflected specularly and diffusely. The fraction of diffuse versus specular radiation that is transmitted and reflected depends on the diffuse fraction for the wall. The special cases of purely diffuse and purely specular transmission and reflection on semi-transparent walls is presented in the following sections.

Note:

For the case of a zero-thickness wall, the wall material's refractive index has no effect on refraction/reflection and only the refractive indices of the adjoining mediums are considered.

If the semi-transparent wall has thickness, then the thickness and the absorption coefficient determine the absorptivity of the 'thin' wall. If either the wall thickness or absorption coefficient is set to 0, then the wall has no absorptivity. Although incident radiation can be absorbed in a semi-transparent wall that has thickness, the default is that the absorbed radiation flux does not affect the energy equation, which can result in an energy imbalance and possibly an unexpected temperature field. The exception is when shell conduction is used (available in 3D only). With shell conduction there is full correspondence between energy and radiation. If the wall is expected to have significant absorption/emission, then it may be better to model the thickness explicitly with solid cells, where practical. Ansys Fluent does not include emission from the surface of semi-transparent walls (due to defined internal emissivity), except when a temperature boundary condition is defined.

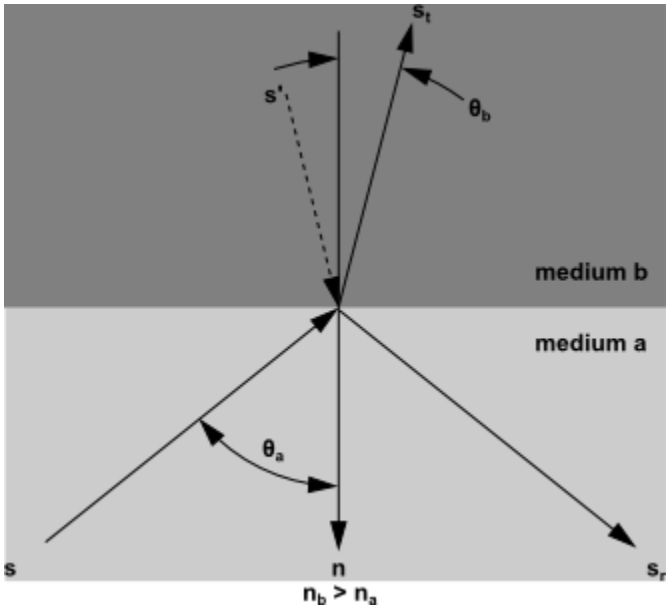
Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

5.3.6.7.2. Specular Semi-Transparent Walls

Consider the special case for a semi-transparent wall, when the diffuse fraction f_d is equal to 0 and all of the transmitted and reflected radiant energy at the semi-transparent wall is purely specular.

Figure 5.10: Reflection and Refraction of Radiation at the Interface Between Two Semi-Transparent Media (p. 209) shows a ray traveling from a semi-transparent medium a with refractive index n_a to a semi-transparent medium b with a refractive index n_b in the direction $\vec{s}$. Surface a of the interface is the side that faces medium a; similarly, surface b faces medium b. The interface normal $\vec{n}$ is assumed to point into side a. We distinguish between the intensity $I_a(\vec{s})$, the intensity in the direction $\vec{s}$ on side a of the interface, and the corresponding quantity on the side b, $I_b(\vec{s})$.

Figure 5.10: Reflection and Refraction of Radiation at the Interface Between Two Semi-Transparent Media



A part of the energy incident on the interface is reflected, and the rest is transmitted. The reflection is specular, so that the direction of reflected radiation is given by

$$\vec{s}_r = \vec{s} - 2(\vec{s}\cdot\vec{n})\,\vec{n} \tag{5.116}$$

The radiation transmitted from medium a to medium b undergoes refraction. The direction of the transmitted energy, $\vec{s}_t$, is given by Snell's law:

$$\sin\theta_b = \frac{n_a}{n_b}\sin\theta_a \tag{5.117}$$

where θ_a is the angle of incidence and θ_b is the angle of transmission, as shown in Figure 5.10: Reflection and Refraction of Radiation at the Interface Between Two Semi-Transparent Media (p. 209). We also define the direction

$$\vec{s}' = \vec{s}_t - 2(\vec{s}_t\cdot\vec{n})\,\vec{n} \tag{5.118}$$

shown in Figure 5.10: Reflection and Refraction of Radiation at the Interface Between Two Semi-Transparent Media (p. 209).

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

The interface reflectivity on side a [444] (p. 1078)

$$r_a(\vec{s}) = \frac{1}{2}\left(\frac{n_a\cos\theta_b - n_b\cos\theta_a}{n_a\cos\theta_b + n_b\cos\theta_a}\right)^2 + \frac{1}{2}\left(\frac{n_a\cos\theta_a - n_b\cos\theta_b}{n_a\cos\theta_a + n_b\cos\theta_b}\right)^2 \tag{5.119}$$

represents the fraction of incident energy transferred from $\vec{s}$ to $\vec{s}_r$.

The boundary intensity $I_{w,a}(\vec{s}_r)$ in the outgoing direction $\vec{s}_r$ on side a of the interface is determined from the reflected component of the incoming radiation and the transmission from side b. Thus

$$I_{w,a}(\vec{s}_r) = r_a(\vec{s})I_{w,a}(\vec{s}) + \tau_b(\vec{s}')I_{w,b}(\vec{s}') \tag{5.120}$$

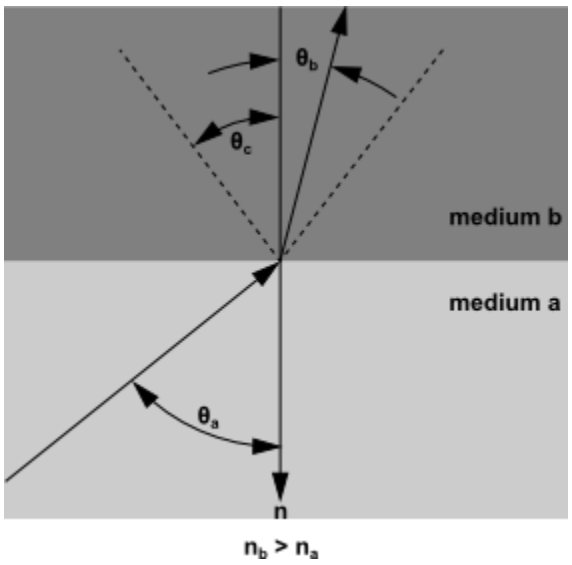
where $\tau_b(\vec{s}')$ is the transmissivity of side b in direction $\vec{s}'$. Similarly, the outgoing intensity in the direction $\vec{s}_t$ on side b of the interface, $I_{w,b}(\vec{s}_t)$, is given by

$$I_{w,b}(\vec{s}_t) = r_b(\vec{s}')I_{w,b}(\vec{s}') + \tau_a(\vec{s})I_{w,a}(\vec{s}) \tag{5.121}$$

For the case $n_a < n_b$, the energy transmitted from medium a to medium b in the incoming solid angle 2π must be refracted into a cone of apex angle θ_c (see Figure 5.11: Critical Angle θc (p. 210)) where

$$\theta_c = \sin^{-1}\frac{n_a}{n_b} \tag{5.122}$$

Figure 5.11: Critical Angle θc



Similarly, the transmitted component of the radiant energy going from medium b to medium a in the cone of apex angle θ_c is refracted into the outgoing solid angle 2π. For incident angles greater than θ_c, total internal reflection occurs and all the incoming energy is reflected specularly back into medium b. The equations presented above can be applied to the general case of interior semi-transparent walls that is shown in Figure 5.9: DO Radiation on Interior Semi-Transparent Wall (p. 208).

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

When medium b is external to the domain as in the case of an external semi-transparent wall (Figure 5.12: DO Irradiation on External Semi-Transparent Wall (p. 212)), $I_{w,b}\left(\vec{s}'\right)$ is given in Equation 5.120 (p. 210) as a part of the boundary condition inputs. You supply this incoming irradiation flux in terms of its magnitude, beam direction, and the solid angle over which the radiative flux is to be applied. Note that the refractive index of the external medium is assumed to be 1.

5.3.6.7.3. Diffuse Semi-Transparent Walls

Consider the special case for a semi-transparent wall, when the diffuse fraction f_d is equal to 1 and all of the transmitted and reflected radiant energy at the semi-transparent wall is purely diffuse.

In many engineering problems, the semi-transparent interface may be a diffuse reflector. For such a case, the interfacial reflectivity $r\left(\vec{s}\right)$ is assumed independent of $\vec{s}$, and equal to the hemispherically averaged value r_d. For $n=n_a/\ n_b > 1$, $r_{d,a}$ and $r_{d,b}$ are given by [594] (p. 1087)

$$r_{d,a} = 1-\frac{\left(1-r_{d,b}\right)}{n^2} \tag{5.123}$$

$$r_{d,b}= \frac{1}{2}+\frac{(3n+1)(n-1)}{6(n+1)^2}+\frac{n^2(n^2-1)^2}{(n^2+1)^3}\ln\left(\frac{n-1}{n+1}\right) -\frac{2n^3(n^2+2n-1)}{(n^2+1)(n^4-1)}+\frac{8n^4(n^4+1)}{(n^2+1)(n^4-1)^2}\ln\left(n\right) \tag{5.124}$$

The boundary intensity for all outgoing directions on side a of the interface is given by

$$I_{w,a}=\frac{r_{d,a}q_{in,a}+\tau_{d,b}q_{in,b}}{\pi} \tag{5.125}$$

Similarly for side b,

$$I_{w,b}=\frac{r_{d,b}q_{in,b}+\tau_{d,a}q_{in,a}}{\pi} \tag{5.126}$$

where

$$q_{in,a}=-\int_{4\pi} I_{w,a}\vec{s}\cdot\vec{n}d\Omega, \vec{s}\cdot\vec{n}<0 \tag{5.127}$$

$$q_{in,b}=\int_{4\pi} I_{w,b}\vec{s}\cdot\vec{n}d\Omega, \vec{s}\cdot\vec{n}\geq 0 \tag{5.128}$$

When medium b is external to the domain as in the case of an external semi-transparent wall (Figure 5.12: DO Irradiation on External Semi-Transparent Wall (p. 212)), $q_{in,b}$ is given as a part of the boundary condition inputs. You supply this incoming irradiation flux in terms of its magnitude, beam direction, and the solid angle over which the radiative flux is to be applied. Note that the refractive index of the external medium is assumed to be 1.

5.3.6.7.4. Partially Diffuse Semi-Transparent Walls

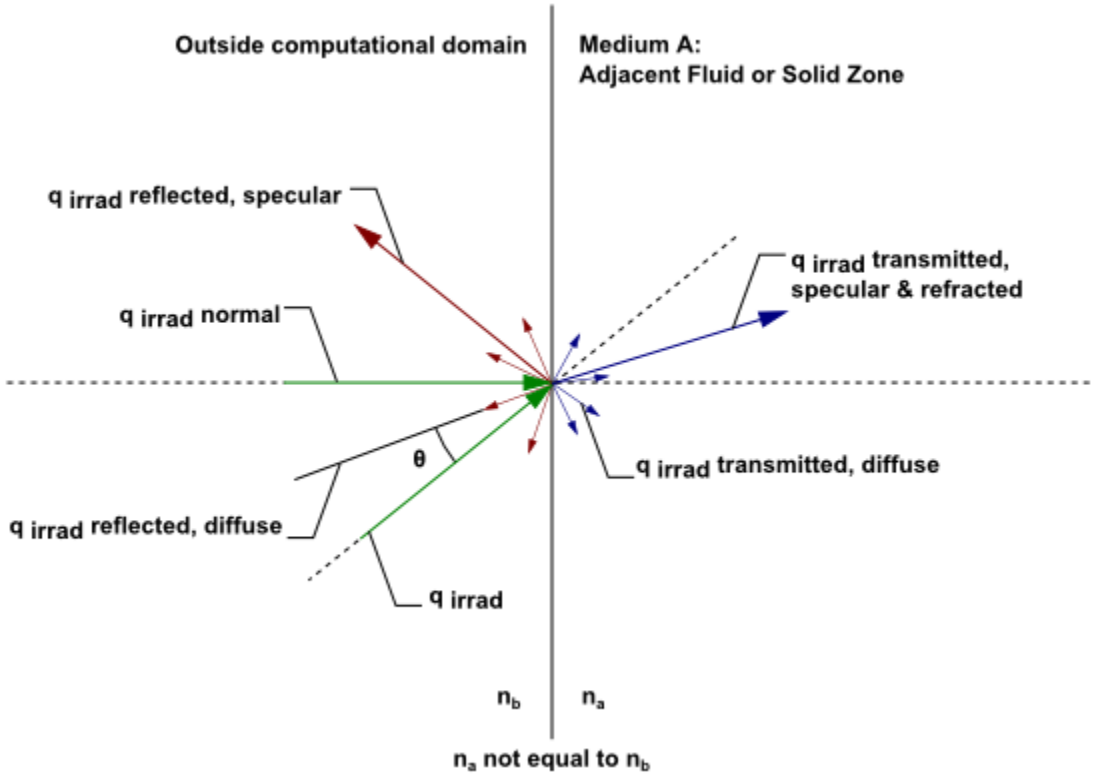
When the diffuse fraction f_d that you enter for a semi-transparent wall is between 0 and 1, the wall is partially diffuse and partially specular. In this case, Ansys Fluent includes the reflective and transmitted radiative flux contributions from both diffuse and specular components to the defining equations.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

5.3.6.7.5. Semi-Transparent Exterior Walls

Figure 5.12: DO Irradiation on External Semi-Transparent Wall (p. 212) shows the general case of an irradiation beam q_{irrad} applied to an exterior semi-transparent wall with zero-thickness and a nonzero absorption coefficient for the material property. Refer to the previous section for the radiation effects of wall thickness on semi-transparent walls.

Figure 5.12: DO Irradiation on External Semi-Transparent Wall



An irradiation flux passes through the semi-transparent wall from outside the computational domain (Figure 5.12: DO Irradiation on External Semi-Transparent Wall (p. 212)) into the adjacent fluid or solid medium a. The transmitted radiation can be refracted (bent) and dispersed specularly and diffusely, depending on the refractive index and the diffuse fraction that you provide as a boundary condition input. Note that there is a reflected component of q_{irrad} when the refractive index of the wall (n_b) is not equal to 1, as shown.

There is an additional flux beyond q_{irrad} that is applied when the **Mixed** or **Radiation** wall boundary conditions are selected in the **Thermal** tab. This external flux at the semi-transparent wall is computed by Ansys Fluent as

$$Q_{ext}=\varepsilon_{external}\sigma T_{rad}^{4} \tag{5.129}$$

The fraction of the above energy that will enter into the domain depends on the transmissivity of the semi-transparent wall under consideration. Note that this energy is distributed across the solid angles (that is, similar treatment as diffuse component.)

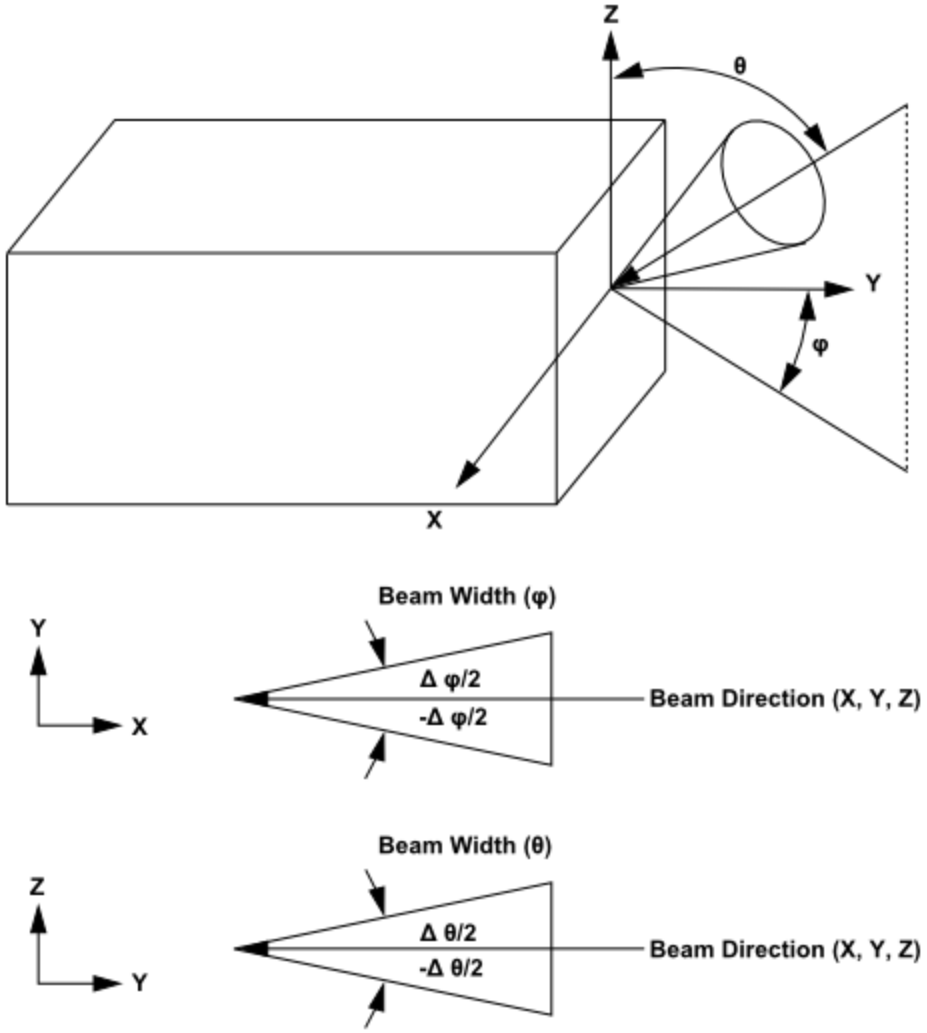
Incident radiation can also occur on external semi-transparent walls. Refer to the previous discussion on interior walls for details, since the radiation effects are the same.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

The irradiation beam is defined by the magnitude, beam direction, and beam width that you supply. The irradiation magnitude is specified in terms of an incident radiant heat flux (W/m^2). Beam width is specified as the solid angle over which the irradiation is distributed (that is, the beam θ and φ extents). The default beam width in Ansys Fluent is $1e-6$ degrees which is suitable for collimated beam radiation. Beam direction is defined by the vector of the centroid of the solid angle. If you select the feature **Apply Direct Irradiation Parallel to the Beam** in the **Wall** boundary condition dialog box, then you supply q_{irrad} for irradiation (Figure 5.12: DO Irradiation on External Semi-Transparent Wall (p. 212)) and Ansys Fluent computes and uses the surface normal flux $q_{irrad,normal}$ in its radiation calculation. If this feature is not checked, then Fluent automatically applies the source you specify, q_{irrad} normal to the boundary such that $q_{irrad} = q_{irrad,normal}$.

Figure 5.13: Beam Width and Direction for External Irradiation Beam (p. 213) shows a schematic of the beam direction and beam width for the irradiation beam. You provide these inputs (in addition to irradiation magnitude) as part of the boundary conditions for a semi-transparent wall.

Figure 5.13: Beam Width and Direction for External Irradiation Beam



The irradiation beam can be refracted in medium a depending on the refractive index that is specified for the particular fluid or solid zone material.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

5.3.6.7.6. Limitations

Where shell conduction is not active, there is only limited support for absorbing and emitting semi-transparent thin walls. In cases with significant emission or absorption of radiation in a participating solid material, such as the absorption of long wavelength radiation in a glass window, the use of semi-transparent thin walls can result in the prediction of unphysical temperatures in the numerical solution. In a 3-dimensional model this can be overcome by activating the shell conduction option for the respective thin wall. Otherwise, where possible, it is advisable to represent the solid wall thickness explicitly with one or more layers of cells across the wall thickness.

5.3.6.7.7. Solid Semi-Transparent Media

The discrete ordinates radiation model allows you to model a solid zone that has adjacent fluid or solid zones on either side as a "semi-transparent" medium. This is done by designating the solid zone to participate in radiation as part of the boundary condition setup. Modeling a solid zone as a semi-transparent medium allows you to obtain a detailed temperature distribution inside the semi-transparent zone since Ansys Fluent solves the energy equation on a per-cell basis for the solid and provides you with the thermal results. By default, however, the DO equations are solved in fluid zones, but not in any solid zones. Therefore, you will need to specify the solid zone as participating in radiation in the **Solid** dialog box as part of the boundary condition setup.

5.3.6.8. Boundary Condition Treatment at Specular Walls and Symmetry Boundaries

At specular walls and symmetry boundaries, the direction of the reflected ray $\vec{s}_r$ corresponding to the incoming direction $\vec{s}$ is given by Equation 5.116 (p. 209). Furthermore,

$$I_w(\vec{s}_r) = I_w(\vec{s}) \tag{5.130}$$

5.3.6.9. Boundary Condition Treatment at Periodic Boundaries

When rotationally periodic boundaries are used, it is important to use pixelation in order to ensure that radiant energy is correctly transferred between the periodic and shadow faces. A pixelation between 3×3 and 10×10 is recommended.

5.3.6.10. Boundary Condition Treatment at Flow Inlets and Outlets

For opaque flow inlet and outlets, the treatment is described in Boundary Condition Treatment for the DTRM at Flow Inlets and outlets (p. 197).

For transparent flow inlet and outlets, the boundary behaves similarly to an external semi-transparent wall, however, the irradiation flux passes through the transparent flow boundary from outside the computational domain into the adjacent fluid zone without getting reflected, absorbed or refracted. For details, see Semi-Transparent Exterior Walls (p. 212).

5.3.7. Surface-to-Surface (S2S) Radiation Model Theory

The surface-to-surface radiation model can be used to account for the radiation exchange in an enclosure of gray-diffuse surfaces. The energy exchange between two surfaces depends in part on their

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

size, separation distance, and orientation. These parameters are accounted for by a geometric function called a "view factor".

The main assumption of the S2S model is that any absorption, emission, or scattering of radiation can be ignored; therefore, only "surface-to-surface" radiation need be considered for analysis.

For information about setting up the model, see Setting Up the S2S Model in the User's Guide.

5.3.7.1. Gray-Diffuse Radiation

Ansys Fluent's S2S radiation model assumes the surfaces to be gray and diffuse. Emissivity and absorptivity of a gray surface are independent of the wavelength. Also, by Kirchoff's law [444] (p. 1078), the emissivity equals the absorptivity ($\varepsilon=\alpha$). For a diffuse surface, the reflectivity is independent of the outgoing (or incoming) directions.

The gray-diffuse model is what is used in Ansys Fluent. Also, as stated earlier, for applications of interest, the exchange of radiative energy between surfaces is virtually unaffected by the medium that separates them. Thus, according to the gray-body model, if a certain amount of radiant energy (E) is incident on a surface, a fraction (ρE) is reflected, a fraction (αE) is absorbed, and a fraction (τE) is transmitted. Since for most applications the surfaces in question are opaque to thermal radiation (in the infrared spectrum), the surfaces can be considered opaque. The transmissivity, therefore, can be neglected. It follows, from the conservation of energy, that $\alpha+\rho=1$, since $\alpha=\varepsilon$ (emissivity), and $\rho=1-\varepsilon$.

5.3.7.2. The S2S Model Equations

The energy flux leaving a given surface is composed of directly emitted and reflected energy. The reflected energy flux is dependent on the incident energy flux from the surroundings, which then can be expressed in terms of the energy flux leaving all other surfaces. The energy leaving from surface k is

$$q_{out,k}=\varepsilon_k\sigma T_k^4+\rho_k q_{in,k} \tag{5.131}$$

where $q_{out,k}$ is the energy flux leaving the surface, ε_k is the emissivity, σ is the Stefan-Boltzmann constant, and $q_{in,k}$ is the energy flux incident on the surface from the surroundings.

The amount of incident energy upon a surface from another surface is a direct function of the surface-to-surface "view factor," F_{jk}. The view factor F_{jk} is the fraction of energy leaving surface j that is incident on surface k. The surfaces used in the calculation of a view factor can be mesh faces or (for 3D cases only) clusters of faces; see Clustering and View Factors (p. 216) for details about clusters. The incident energy flux $q_{in,k}$ can be expressed in terms of the energy flux leaving all other surfaces as

$$A_k q_{in,k}=\sum_{j=1}^{N} A_j q_{out,j} F_{jk} \tag{5.132}$$

where A_k is the area of surface k and F_{jk} is the view factor between surface k and surface j. For N surfaces, using the view factor reciprocity relationship gives

$$A_j F_{jk}=A_k F_{kj} \text{ for } j=1,2,3,...N \tag{5.133}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

so that

$$q_{in,k} = \sum_{j=1}^{N} F_{kj} q_{out,j} \tag{5.134}$$

Therefore,

$$q_{out,k} = \varepsilon_k \sigma T_k^4 + \rho_k \sum_{j=1}^{N} F_{kj} q_{out,j} \tag{5.135}$$

which can be written as

$$J_k = E_k + \rho_k \sum_{j=1}^{N} F_{kj} J_j \tag{5.136}$$

where J_k represents the energy that is given off (or radiosity) of surface k, and E_k represents the emissive power of surface k. This represents N equations, which can be recast into matrix form as

$$KJ = E \tag{5.137}$$

where K is an $N \times N$ matrix, J is the radiosity vector, and E is the emissive power vector.

Equation 5.137 (p. 216) is referred to as the radiosity matrix equation. The view factor between two finite surfaces i and j is given by

$$F_{ij} = \frac{1}{A_i} \int_{A_i} \int_{A_j} \frac{\cos\theta_i \cos\theta_j}{\pi r^2} \delta_{ij} dA_i dA_j \tag{5.138}$$

where δ_{ij} is determined by the visibility of dA_j to dA_i. $\delta_{ij} = 1$ if dA_j is visible to dA_i and 0 otherwise.

5.3.7.3. Clustering

The S2S radiation model is computationally very expensive when you calculate the radiation and view factors for a large number of surfaces. To reduce the computational time as well as the storage requirement, the number of surfaces is reduced by creating surface "clusters". The surface clusters are made by starting from a face and adding its neighbors and their neighbors until a specified number of faces per surface cluster is collected.

5.3.7.3.1. Clustering and View Factors

By default, view factors are calculated using a face to face basis, in which clustering is used in a limited way only. The boundary faces act as the surfaces for the view factor calculation, and then a cluster view factor is obtained by taking the area-weighted average of the view factors of the faces within the cluster.

For 3D cases, you have the option of using the cluster to cluster basis instead of the face to face basis, which can reduce the computational expense and storage requirements. In this approach, Ansys Fluent internally creates polygon faces by combining all of the faces from non-polyhedral cells in each cluster, and these are then used as the surfaces for the view factor calculation. The `.s2s` file will contain the connectivity information of the cluster itself instead of the individual faces in the cluster, and the coarser, polyhedral mesh that is written requires less disk space. Note that accuracy can be impacted by the use of the cluster to cluster basis, because of the following reasons:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

- Clusters are assumed to be planar for view factor computation, even though they may not be exactly planar. This discrepancy reduces the accuracy, especially as the surface becomes more convex / concave, and as the number of faces per surface clusters for the boundary zones increases.
- The option of dividing faces into subfaces as part of the hemicube algorithm (which increases accuracy) is not available with the cluster to cluster basis.

5.3.7.3.2. Clustering and Radiosity

The radiosity, J, is calculated for the surface clusters. These values are then distributed to the faces in the clusters to calculate the wall temperatures. Since the radiation source terms are highly nonlinear (proportional to the fourth power of temperature), care must be taken to calculate the average temperature of the surface clusters and distribute the flux and source terms appropriately among the faces forming the clusters.

The surface cluster temperature is obtained by area averaging as shown in the following equation:

$$T_{sc} = \left(\frac{\sum_f A_f T_f^4}{\sum A_f} \right)^{1/4} \tag{5.139}$$

where T_{sc} is the temperature of the surface cluster, and A_f and T_f are the area and temperature of face f. The summation is carried over all faces of a surface cluster.

5.3.8. Monte Carlo (MC) Radiation Model Theory

The Monte Carlo radiation model simulates the underlying processes that govern the system of interest (that is, the physical interactions between photons and their environment). A photon is selected from a photon source and tracked through the system until its weight falls below some minimum, at which point it "dies". Each time the photon experiences an "event", (for example, a surface intersection, scattering, or absorption), the physical quantities of interest are updated. This process generates a complete "history" of that photon in the system. Many photon histories need to be generated to get good estimates of the physical quantities of interest in a system. Photon sources are selected (that is, "sampled") on the basis of emitted radiation, each band being treated independently for non-gray models.

This section provides details about the equations used in the MC model. For information about setting up the model, see Setting Up the MC Model in the User's Guide.

5.3.8.1. The MC Model Equations

For the radiative transfer equation (RTE), the Monte Carlo model assumes that the intensity is proportional to the differential angular flux of photons and treats the radiation field as a photon gas. For this gas, a is the probability per unit length that a photon is absorbed at a given frequency. Therefore, the mean radiation intensity, I is proportional to the distance traveled by a photon in unit volume at $\vec{r}$, in unit time.

Similarly, the spectral radiative heat flux q_{in} is proportional to the rate of incidence of photons on the surface at $\vec{r}$, because volumetric absorption is proportional to the rate of absorption of photons.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

By following a typical selection of photons and tallying, in each volume cell, the distance traveled, you can obtain the mean total intensity.

By following a typical selection of photons and tallying, in each volume cell, the distance times the absorption coefficient, you can obtain the mean total absorbed intensity.

By following a typical selection of photons and tallying, in each volume cell, the distance times the scattering coefficient, you can obtain the mean total scattered intensity.

By also tallying the number of photons incident on a surface and this number times the emissivity, you can obtain the mean total radiative flux and the mean absorbed flux.

Note that no discretization of the spectrum is required, because differential quantities are not usually important for heat transfer calculations. Providing that the spectral (multiband) selection is done properly, the Monte Carlo tallying automatically integrates over the spectrum. Boundary conditions for the non-gray DO model are applied on a band basis. The treatment within a band is the same as that for the gray DO model.

5.3.8.1.1. Monte Carlo Solution Accuracy

The Monte Carlo radiation model generates photons in a stochastic (random) manner and will therefore produce *speckled* results if the target number of histories is relatively small. Increasing the target number of histories produces a smoother and more accurate solution, but at the expense of higher computation effort.

5.3.8.2. Boundary Condition Treatment for the MC Model

The treatment at boundaries is similar to that of the DO model for gray radiation. For further details, see the sections on boundary condition treatment for the DO model.

Note:

External walls with boundary source and shell conduction is not supported (shell conduction is disabled if you enable both).

5.3.9. Radiation in Combusting Flows

5.3.9.1. The Weighted-Sum-of-Gray-Gases Model

The weighted-sum-of-gray-gases model (WSGGM) is a reasonable compromise between the over-simplified gray gas model and a complete model which takes into account particular absorption bands. The basic assumption of the WSGGM is that the total emissivity over the distance s can be presented as

$$\varepsilon = \sum_{i=0}^{I} a_{\varepsilon,i}(T)\left(1 - e^{-\kappa_i p s}\right) \tag{5.140}$$

where $a_{\varepsilon,i}$ is the emissivity weighting factor for the i^{th} fictitious gray gas, the bracketed quantity is the i^{th} fictitious gray gas emissivity, κ_i is the absorption coefficient of the i^{th} gray gas, p is the sum of the partial pressures of all absorbing gases, and s is the path length. For $a_{\varepsilon,i}$ and κ_i Ansys

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Fluent uses values obtained from [121] (p. 1060) and [608] (p. 1088). These values depend on gas composition, and $a_{\varepsilon,i}$ also depend on temperature. When the total pressure is not equal to 1 atm, scaling rules for κ_i are used (see Equation 5.147 (p. 220)).

The absorption coefficient for $i=0$ is assigned a value of zero to account for windows in the spectrum between spectral regions of high absorption ($\sum_{i=1}^{I} a_{\varepsilon,i}<1$) and the weighting factor for $i=0$ is evaluated from [608] (p. 1088):

$$a_{\varepsilon,0}=1-\sum_{i=1}^{I} a_{\varepsilon,i} \tag{5.141}$$

The temperature dependence of $a_{\varepsilon,i}$ can be approximated by any function, but the most common approximation is

$$a_{\varepsilon,i}=\sum_{j=1}^{J} b_{\varepsilon,i,j} T^{j-1} \tag{5.142}$$

where $b_{\varepsilon,i,j}$ are the emissivity gas temperature polynomial coefficients. The coefficients $b_{\varepsilon,i,j}$ and κ_i are estimated by fitting Equation 5.140 (p. 218) to the table of total emissivities, obtained experimentally [121] (p. 1060), [138] (p. 1060), [608] (p. 1088).

The absorptivity α of the radiation from the wall can be approximated in a similar way [608] (p. 1088), but, to simplify the problem, it is assumed that $\varepsilon=\alpha$ [443] (p. 1078). This assumption is justified unless the medium is optically thin and the wall temperature differs considerably from the gas temperature.

Since the coefficients $b_{\varepsilon,i,j}$ and κ_i are slowly varying functions of ps and T, they can be assumed constant for a wide range of these parameters. In [608] (p. 1088) these constant coefficients are presented for different relative pressures of the CO_2 and H_2O vapor, assuming that the total pressure p_T is 1 atm. The values of the coefficients shown in [608] (p. 1088) are valid for $0.001 \le ps \le 10.0$ atm-m and $600 \le T \le 2400$ K. For $T>2400$ K, coefficient values suggested by [121] (p. 1060) are used. If $\kappa_i ps \ll 1$ for all i, Equation 5.140 (p. 218) simplifies to

$$\varepsilon=\sum_{i=0}^{I} a_{\varepsilon,i}\kappa_i ps \tag{5.143}$$

Comparing Equation 5.143 (p. 219) with the gray gas model with absorption coefficient a, it can be seen that the change of the radiation intensity over the distance s in the WSGGM is exactly the same as in the gray gas model with the absorption coefficient

$$a=\sum_{i=0}^{I} a_{\varepsilon,i}\kappa_i p \tag{5.144}$$

which does not depend on s. In the general case, a is estimated as

$$a=-\frac{\ln\left(1-\varepsilon\right)}{s} \tag{5.145}$$

where the emissivity ε for the WSGGM is computed using Equation 5.140 (p. 218). a as defined by Equation 5.145 (p. 219) depends on s, reflecting the non-gray nature of the absorption of thermal radiation in molecular gases. In Ansys Fluent, Equation 5.144 (p. 219) is used when $s \le 10^{-4}$ m and

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Equation 5.145 (p. 219) is used for $s > 10^{-4}$ m. Note that for $s \approx 10^{-4}$ m, the values of a predicted by Equation 5.144 (p. 219) and Equation 5.145 (p. 219) are practically identical (since Equation 5.145 (p. 219) reduces to Equation 5.144 (p. 219) in the limit of small s).

Ansys Fluent allows you to set s equal to the mean beam length. You can specify the mean beam length or have Ansys Fluent compute it according to the following equation from Siegel [593] (p. 1087).

$$s = 3.6\frac{V}{A} \tag{5.146}$$

where V is the fluid volume and A is the total surface area of the fluid boundaries (walls, inlets, outlets; not symmetry).

See Inputs for a Composition-Dependent Absorption Coefficient in the User's Guide for details about setting properties for the WSGGM.

Important:

The WSGGM is implemented in a gray approach. If the WSGGM is used with a non-gray model, the absorption coefficient will be the same in all bands. Use `DEFINE_GRAY_BAND_ABS_COEFF` to change the absorption coefficient per band or per gray gas.

5.3.9.1.1. When the Total (Static) Gas Pressure is Not Equal to 1 atm

The WSGGM, as described above, assumes that p_{tot}—the total (static) gas pressure—is equal to 1 atm. In cases where p_{tot} is not unity (for example, combustion at high temperatures), scaling rules suggested in [158] (p. 1062) are used to introduce corrections. When $p_{tot} < 0.9$ atm or $p_{tot} > 1.1$ atm, the values for κ_i in Equation 5.140 (p. 218) and Equation 5.144 (p. 219) are rescaled:

$$\kappa_i \rightarrow \kappa_i p_{tot}^m \tag{5.147}$$

where m is a non-dimensional value obtained from [158] (p. 1062), which depends on the partial pressures and temperature T of the absorbing gases, as well as on p_{tot}.

5.3.9.2. The Effect of Soot on the Absorption Coefficient

When soot formation is computed, Ansys Fluent can include the effect of the soot concentration on the radiation absorption coefficient. The generalized soot model estimates the effect of the soot on radiative heat transfer by determining an effective absorption coefficient for soot. The absorption coefficient of a mixture of soot and an absorbing (radiating) gas is then calculated as the sum of the absorption coefficients of pure gas and pure soot:

$$a_{s+g} = a_g + a_s \tag{5.148}$$

where a_g is the absorption coefficient of gas without soot (obtained from the WSGGM) and

$$a_s = b_1 \rho_g Y_{soot}[1 + b_T(T - 2000)] \tag{5.149}$$

with

$$b_1 = 1232.4 \text{m}^2 / \text{ kg}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

and

$$b_T \approx 4.8 \times 10^{-4} \mathrm{K}^{-1}$$

ρ_g is the gas mixture density in kg/m^3 and Y_{soot} is the soot mass fraction.

The coefficients b_1 and b_T were obtained [569] (p. 1086) by fitting Equation 5.149 (p. 220) to data based on the Taylor-Foster approximation [645] (p. 1090) and data based on the Smith et al. approximation [608] (p. 1088).

See Radiation Properties and Using the Soot Models in the User's Guide for information about including the soot-radiation interaction effects.

5.3.9.3. The Effect of Particles on the Absorption Coefficient

Ansys Fluent can also include the effect of discrete phase particles on the radiation absorption coefficient, provided that you are using either the P-1 or the DO model. When the P-1 or DO model is active, radiation absorption by particles can be enabled. The particle emissivity, reflectivity, and scattering effects are then included in the calculation of the radiative heat transfer. See Setting Material Properties for the Discrete Phase in the User's Guide for more details on the input of radiation properties for the discrete phase.

5.3.10. Choosing a Radiation Model

For certain problems one radiation model may be more appropriate than the others. When deciding which radiation model to use, consider the following:

- Optical thickness: The optical thickness aL is a good indicator of which model to use in your problem. Here, L is an appropriate length scale for your domain. For flow in a combustor, for example, L is the diameter of the combustion chamber. If $aL \gg 1$, your best alternatives are the P-1 and Rosseland models. The P-1 model should typically be used for optical thicknesses >1. For optical thickness >3, the Rosseland model is cheaper and more efficient. For high optical thickness cases, a second-order discretization scheme for the DO model is recommended. The DTRM, DO, and MC models work across the full range of optical thicknesses, but are substantially more expensive to use. Consequently, you should use the "thick-limit" models, P-1 and Rosseland, if the problem allows it. For optically thin problems ($aL<1$), the DTRM,DO, and MC models, only, are appropriate.

- Scattering and emissivity: The P-1, Rosseland, and DO models account for scattering, while the DTRM neglects it. Since the Rosseland model uses a temperature slip condition at walls, it is insensitive to wall emissivity.

- Particulate effects: Only the P-1 and DO models account for exchange of radiation between gas and particulates (see Equation 5.68 (p. 192)).

- Semi-transparent walls:(interior and exterior) The DO and MC models allow you to model semi-transparent walls of various types (for example, glass). The MC model does allow the semi-transparent boundary condition in the case of coupled walls.

- Specular walls: The DO and MC models allow for specular reflection (for example, for dust-free mirror).

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

- Partially specular walls: The DO and MC models allow for partially specular reflection (for example, dusty mirror).
- Non-gray radiation: Only the P-1, DO, and MC models allow you to compute non-gray radiation using a gray-band model.
- Localized heat sources: In problems with localized sources of heat, the P-1 model may over-predict the radiative fluxes. The DO model is probably the best suited for computing radiation for this case, although the DTRM, with a sufficiently large number of rays, is also acceptable.
- Enclosure radiative transfer with non-participating media: The surface-to-surface (S2S) model is suitable for this type of problem. The radiation models used with participating media may, in principle, be used to compute the surface-to-surface radiation, but they are not always efficient.

5.3.10.1. External Radiation

If you need to include radiative heat transfer from the exterior of your physical model, you can include an external radiation boundary condition in your model (for details, see Thermal Boundary Conditions at Walls in the User's Guide). If you are not concerned with radiation within the domain, this boundary condition can be used without activating one of the radiation models.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Chapter 6: Heat Exchangers

Many engineering systems, including power plants, climate control, and engine cooling systems typically contain tubular heat exchangers. However, for most engineering problems, it is impractical to model individual fins and tubes of a heat exchanger core. In principle, heat exchanger cores introduce a pressure drop to the primary fluid stream and transfer heat from or to a second fluid (such as a coolant), referred to here as the auxiliary fluid.

In Ansys Fluent, lumped-parameter models are used to account for the pressure loss and auxiliary fluid heat rejection. Ansys Fluent provides two heat exchanger models: the macro (ungrouped and grouped) models and the dual cell model. The macro model allows you to choose between two heat transfer models, namely the **simple-effectiveness-model** and the number-of-transfer-units (NTU) model. The models can be used to compute auxiliary fluid inlet temperature for a fixed heat rejection or total heat rejection for a fixed auxiliary fluid inlet temperature. For the **simple-effectiveness-model**, the auxiliary fluid may be single-phase or two-phase. The dual cell model uses the NTU method for heat transfer calculations. This model allows the solution of auxiliary flow on a separate mesh (other than the primary fluid mesh), unlike the macro model, where the auxiliary flow is modeled as 1D flow. The dual cell model also offers more flexibility as far as the shape of the heat exchanger is concerned, and overcomes some of the major limitations present in the macro model.

For more information about using the heat exchanger models, see Modeling Heat Exchangers in the User's Guide.

The following sections contain information about the theory behind the heat exchanger models:

6.1. The Macro Heat Exchanger Models

6.2. The Dual Cell Model

6.1. The Macro Heat Exchanger Models

The following sections contain information about the theory behind the macro heat exchanger models:

6.1.1. Overview of the Macro Heat Exchanger Models

6.1.2. Restrictions of the Macro Heat Exchanger Models

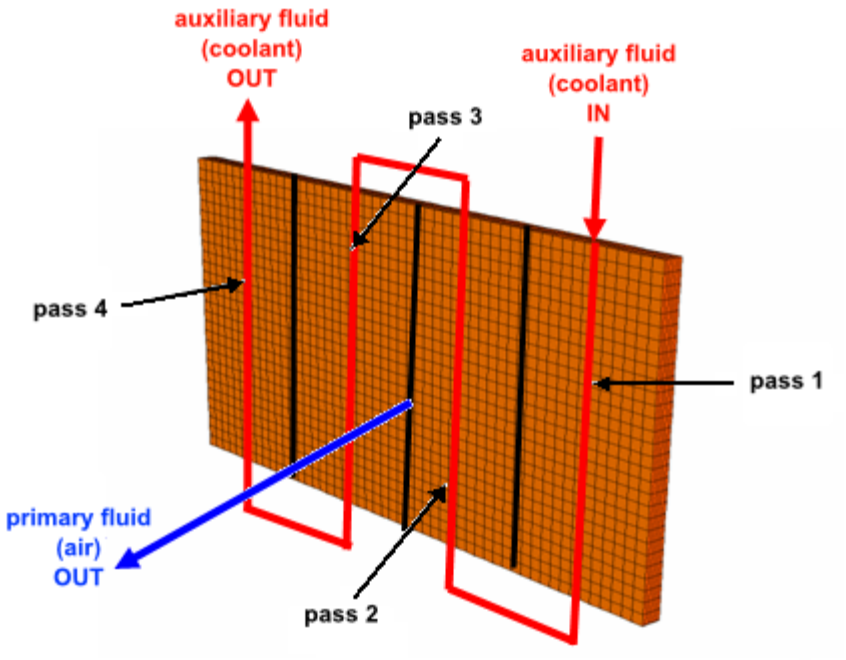
6.1.3. Macro Heat Exchanger Model Theory

6.1.1. Overview of the Macro Heat Exchanger Models

A heat exchanger consists of a primary fluid (for example, air) that comes into contact with one or more passes of an auxiliary fluid (for example, a coolant) that has a different temperature.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Figure 6.1: An Example of a Four-Pass Heat Exchanger



In a typical heat exchanger core, the auxiliary fluid temperature is stratified in the direction of the auxiliary fluid flow. As a result, heat rejection is not constant over the entire core. In Ansys Fluent, the fluid zone representing the heat exchanger core is subdivided into macroscopic cells or *macros* along the auxiliary fluid path, as in Figure 6.2: Core Discretized into 3x4x2 Macros (p. 225). In this figure, the core is discretized into 3x4x2 macros. This configuration consists of two passes, each pass having four rows and three columns of macros. The auxiliary fluid inlet temperature to each macro is computed and then subsequently used to compute the heat rejection from each macro. This approach provides a realistic heat rejection distribution over the heat exchanger core.

To use the heat exchanger models, you must define one or more fluid zone(s) to represent the heat exchanger core. Typically, the fluid zone is sized to the dimension of the core itself. As part of the setup procedure, you will define the auxiliary fluid path, the number of macros, and the physical properties and operating conditions of the core (pressure drop parameters, heat exchanger effectiveness, auxiliary fluid flow rate, and so on).

You can also combine several fluid zones as a single heat exchanger group. In this situation each fluid zone acts as a separate heat exchanger core, and the auxiliary fluid mass flow rate of the heat exchanger group is divided among the zones in the ratio of the respective volumes. For the purpose of auxiliary fluid flow, heat exchanger groups can also be connected in series. In addition, a heat exchanger group can have an auxiliary fluid pressure drop (for example, for pressure dependent properties) and a supplementary auxiliary fluid stream entering or leaving it. For more information on heat exchanger groups, see Using the Grouped Macro Heat Exchanger Model.

The heat exchanger models were designed for "compact" heat exchangers, implying that the primary fluid side flow is unidirectional. The auxiliary fluid is assumed to flow through a large number of parallel tubes, which can optionally double back in a serpentine pattern to create a number of "passes".

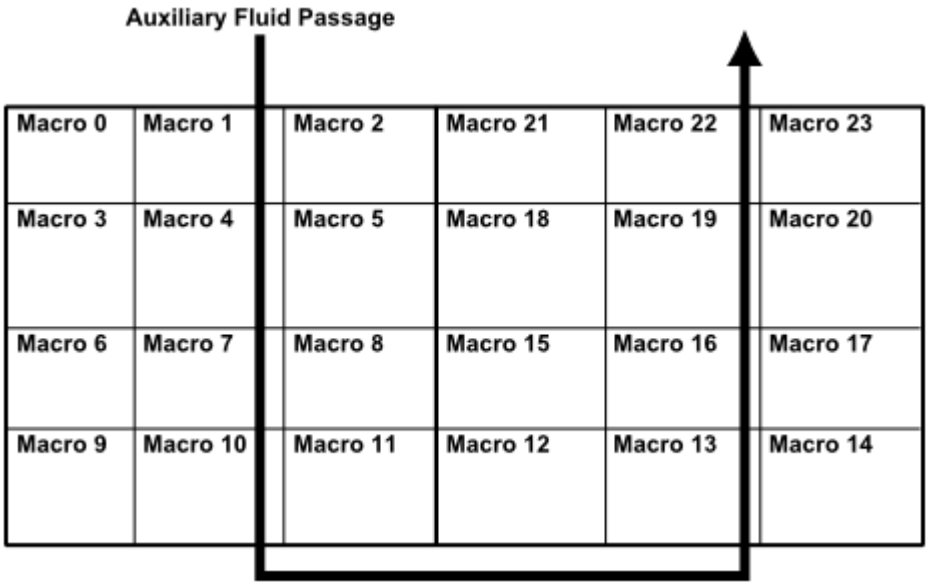
Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

You can independently choose the principal auxiliary fluid flow direction, the pass-to-pass direction and the external primary fluid flow direction.

Important:

It is highly recommended that the free-form **Tet** mesh is not used in the macro heat exchanger model. Instead, evenly distributed **Hex/Wedge** cells should be used for improved accuracy and a more robust solution process.

Figure 6.2: Core Discretized into 3x4x2 Macros



6.1.2. Restrictions of the Macro Heat Exchanger Models

The following restrictions exist for the macro heat exchanger models:

- The core must be a 3D mesh with a cross-section that is approximately rectangular in shape.
- The primary fluid streamwise direction (see Equation 6.1 (p. 227)) must be aligned with one of the three orthogonal axes defined by the rectangular core.
- The pass-to-pass plane must be perpendicular to the primary fluid streamwise direction.
- The two dimensions of the pass-to-pass plane can each be discretized into multiple macroscopic cells (macros), but in the direction perpendicular to this plane the macros cannot be subdivided.
- Flow acceleration effects are neglected in calculating the pressure loss coefficient.
- For the **simple-effectiveness-model**, the primary fluid must have a capacity rate that is less than that of the auxiliary fluid.
- Auxiliary fluid phase change cannot be modeled using the **ntu-model**.
- The macro-based method requires that an equal number of cells reside in each macro of equal size and shape.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

- Auxiliary fluid flow is assumed to be 1D.
- The pass width has to be uniform.
- Accuracy is not guaranteed when the mesh is not structured or layered.
- Accuracy is not guaranteed when there is upstream diffusion of temperature at the inlet/outlet of the core.
- Non-conformal meshes cannot be attached to the inlet/outlet of the core. An extra layer has to be created to avoid it.

6.1.3. Macro Heat Exchanger Model Theory

In Ansys Fluent, the heat exchanger core is treated as a fluid zone with momentum and heat transfer. Pressure loss is modeled as a momentum sink in the momentum equation, and heat transfer is modeled as a heat source in the energy equation.

Ansys Fluent provides two heat transfer models: the default **ntu-model** and the **simple-effectiveness-model**. The **simple-effectiveness-model** interpolates the effectiveness from the velocity vs. effectiveness curve that you provide. For the **ntu-model**, Ansys Fluent calculates the effectiveness, ε, from the NTU value that is calculated by Ansys Fluent from the heat transfer data provided by you in tabular format. Ansys Fluent will automatically convert this heat transfer data to a primary fluid mass flow rate vs. NTU curve (this curve will be piecewise linear). This curve will be used by Ansys Fluent to calculate the NTU for macros based on their size and primary fluid flow rate. See Heat Transfer Effectiveness (p. 228) and Heat Rejection (p. 229) for the equations used in these calculations.

The **ntu-model** provides the following features:

- The model can be used to check the heat capacity rate for both the primary and the auxiliary fluid, and takes the lesser of the two for the calculation of heat transfer.
- The model can be used to model heat transfer to the primary fluid from the auxiliary fluid, and vice versa.
- The model can be used to model primary fluid-side reverse flow.
- The model can be used with variable density of the primary fluid.
- The model can be used in either the serial or parallel Ansys Fluent solvers.
- Transient profiles can be used for the auxiliary fluid inlet temperature and for total heat rejection.
- Transient profiles can be used for auxiliary mass flow rates.

The **simple-effectiveness-model** provides the following features:

- The model can be used to model heat transfer from the auxiliary fluid to the primary fluid, and vice versa.
- The auxiliary fluid properties can be a function of pressure and temperature, therefore allowing phase change of the auxiliary fluid.
- The model can be used by serial as well as parallel solvers.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

- The model can be used to make a network of heat exchangers using a heat exchanger group (Using the Grouped Macro Heat Exchanger Model).
- Transient profiles can be used for the auxiliary fluid inlet temperature and for total heat rejection.
- Transient profiles can be used for auxiliary mass flow rates.

6.1.3.1. Streamwise Pressure Drop

In both heat transfer models, pressure loss is modeled using the porous media model in Ansys Fluent. For the dual cell model (The Dual Cell Model (p. 231)), pressure loss is used for both streams, while for the macro model, it is used only for the primary side.

The loss coefficients of the porous media model are computed using curve fitting of the pressure-versus-flow rate data outside of Ansys Fluent, which you will specify for the cell zone conditions. However, in some cases, the data for curve-fitting is not available. The macro model provides an additional means of getting the coefficients if the data is not available. The coefficients can also be automatically computed (and updated) using a known pressure loss coefficient as a function of some geometric parameters, the theory of which is defined below:

$$\Delta p = \frac{1}{2} f \rho_m U^2_{A_{min}} \tag{6.1}$$

where

Δp = streamwise pressure drop

f = streamwise pressure loss coefficient

ρ_m = mean primary fluid density

$U_{A_{min}}$ = primary fluid velocity at the minimum flow area

The pressure loss coefficient is computed from

$$f = \left(K_c + 1 - \sigma^2\right) - \left(1 - \sigma^2 - K_e\right)\frac{v_e}{v_i} + 2\left(\frac{v_e}{v_i} - 1\right) + f_c \frac{A}{A_c}\frac{v_m}{v_i} \tag{6.2}$$

where

σ = ratio of free-flow area to core frontal area

K_c = entrance loss coefficient

K_e = exit loss coefficient

A = primary fluid-side surface area

A_c = heat exchanger core minimum free flow area

f_c = core friction factor

v_e = specific volume at the exit

v_i = specific volume at the inlet

v_m = mean specific volume $\equiv \frac{1}{2}\left(v_e + v_i\right)$

K_c and K_e are empirical quantities obtained from experimental data. You will need to specify these parameters based on graphs that are closest to the heat exchanger configuration that you are setting up [296] (p. 1070), [294] (p. 1070). These parameters are used to set up large resistances in the two

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

non-streamwise directions, effectively forcing the primary fluid flow through the core to be unidirectional.

In Equation 6.2 (p. 227), the core friction factor is defined as

$$f_c = aRe_{min}^{b} \tag{6.3}$$

where

a = core friction coefficient

b = core friction exponent

Re_{min} = Reynolds number for velocity at the minimum flow area

a and b are empirical quantities obtained from experimental data. You will need to specify the core friction coefficient and exponent based on graphs that are closest to the heat exchanger models that you set up [296] (p. 1070), [294] (p. 1070).

The Reynolds number in Equation 6.3 (p. 228) is defined as

$$Re_{min} = \frac{\rho_m U_{A_{min}} D_h}{\mu_m} \tag{6.4}$$

where

ρ_m = mean primary fluid density

μ_m = mean primary fluid viscosity

D_h = hydraulic diameter

$U_{A_{min}}$ = primary fluid velocity at the minimum flow area

For a heat exchanger core, the hydraulic diameter can be defined as

$$D_h = 4L\left(\frac{A_c}{A}\right) \tag{6.5}$$

where L is the flow length of the heat exchanger. If the tubes are normal to the primary fluid flow, then L is the length in the primary fluid flow direction. Note that $U_{A_{min}}$ can be calculated from

$$U_{A_{min}} = \frac{U}{\sigma} \tag{6.6}$$

where U is the primary fluid velocity and σ is the ratio of free-flow area to core frontal area.

6.1.3.2. Heat Transfer Effectiveness

The heat-exchanger effectiveness, ε, is defined as the ratio of actual rate of heat transfer between the primary and auxiliary fluids to the maximum possible rate of heat transfer. The maximum possible heat transfer is given by

$$q_{max} = C_{min}\left|T_{in,primary} - T_{in,auxiliary}\right| \tag{6.7}$$

where $T_{in,primary}$ and $T_{in,auxiliary}$ are the inlet temperatures of the primary and auxiliary fluids and

$$C_{min} = \min\left[\left(\dot{m}c_p\right)_{primary}, \left(\dot{m}c_p\right)_{auxiliary}\right] \tag{6.8}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Thus, the actual rate of heat transfer, q, is defined as

$$q=\varepsilon q_{max} \tag{6.9}$$

The value of ε depends on the heat exchanger geometry and flow pattern (parallel flow, counter flow, cross flow, and so on). Even though the effectiveness of the primary fluid is computed using uniform conditions on the entire heat exchanger core, it will be applied to a small portion of the core represented by a computational cell. This can make it less accurate for some heat exchanger cores, where there is a strong variation in the primary flow.

For the **simple-effectiveness-model**, you provide effectiveness values for the heat exchanger you are simulating, specifying how this quantity changes with velocity. For the **ntu-model**, you provide the heat exchanger performance data (heat rejection / transfer data versus primary flow rate) based on uniform test conditions, and Ansys Fluent calculates the effectiveness of the entire heat exchanger from the ratio of heat capacity and the number of transfer units using the relation

$$\varepsilon=1-\exp\left[-\frac{1}{C_r}N_{tu}^{0.22}\left(1-e^{-C_r N_{tu}^{0.78}}\right)\right] \tag{6.10}$$

where C_r is the ratio of C_{min} to C_{max}.

The heat exchanger performance data should be specified for a number of auxiliary flow rates so that Ansys Fluent can compute the number of transfer units versus the primary fluid flow rate for a number of auxiliary fluid flow rates. This NTU, which is based on the full heat exchanger and uniform conditions, is scaled for each macro using the ratio of their volumes and minimum heat capacities.

For each macro, the primary fluid inlet temperature is calculated using the mass average of the incoming primary fluid temperatures at the boundaries. This automatically takes into account any reverse flow of the primary fluid at the boundaries.

Note that the previous equations are for crossflow (unmixed) conditions; the macro model can only simulate crossflow, as opposed to coflow and counterflow cases.

6.1.3.3. Heat Rejection

Heat rejection is computed for each cell within a macro and added as a source term to the energy equation for the primary fluid flow. Note that heat rejection from the auxiliary fluid to primary fluid can be either positive or negative.

For the **simple-effectiveness-model**, the heat transfer for a given cell is computed from

$$q_{cell}=\varepsilon\left(\dot{m}c_p\right)_{primary}\left(T_{in,auxiliary}-T_{cell}\right) \tag{6.11}$$

where

ε = heat exchanger effectiveness

$\left(\dot{m}c_p\right)_{primary}$ = primary fluid capacity rate (flow rate × specific heat)

$T_{in,auxiliary}$ = auxiliary fluid inlet temperature of macro containing the cell

T_{cell} = cell temperature

For the **simple-effectiveness-model**, the heat rejection from a macro is calculated by summing the heat transfer of all the cells contained within the macro

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$q_{macro} = \sum_{\text{all cells in macro}} q_{cell} \tag{6.12}$$

For the **ntu-model**, the heat transfer for a macro is calculated from

$$q_{macro} = \varepsilon C_{min}\left(T_{in,auxiliary} - T_{in,primary}\right) \tag{6.13}$$

where

ε = macro effectiveness

$T_{in,auxiliary}$ = macro auxiliary fluid inlet temperature

$T_{in,primary}$ = macro primary fluid inlet temperature

For the **ntu-model**, the heat transfer for a given cell is computed from

$$q_{cell} = q_{macro}\frac{V_{cell}}{V_{macro}} \tag{6.14}$$

For both heat exchanger models, the total heat rejection from the heat exchanger core is computed as the sum of the heat rejection from all the macros:

$$q_{total} = \sum_{\text{all macros}} q_{macro} \tag{6.15}$$

The auxiliary fluid inlet temperature to each macro ($T_{in,auxiliary}$ in Equation 6.11 (p. 229) and Equation 6.13 (p. 230)) is computed based on the energy balance of the auxiliary fluid at a previous macro computation. For a given macro,

$$q_{macro} = \left(\dot{m}\right)_{auxiliary}\left(h_{out} - h_{in}\right) \tag{6.16}$$

where h_{in} and h_{out} are the inlet and outlet enthalpies of the auxiliary fluid in the macro. The auxiliary fluid outlet temperature from the macro is calculated as

$$T_{out} = \begin{cases} \frac{h_{out}}{c_{p,auxiliary}} & constant\ specific\ heat\ method \\ f\left(h_{out}, p\right) & UDF\ method \end{cases} \tag{6.17}$$

where

f = user-defined function

p = auxiliary fluid pressure

The values of h_{out} and T_{out} then become the inlet conditions to the next macro.

The first row of macros (Macros 0, 1, and 2 in Figure 6.2: Core Discretized into 3x4x2 Macros (p. 225)) are assumed to be where the auxiliary fluid enters the heat exchanger core. When the fixed total heat rejection from the heat exchanger core is specified, the inlet temperature to the first row of macros is iteratively computed, so that all of the equations are satisfied simultaneously. When a fixed auxiliary fluid inlet temperature is specified, the heat transfer for the first row of macros are used to calculate their exit enthalpy, which becomes the inlet condition for the next row macros. At the end of each pass, the outlet enthalpy of each macro (in the last row) is mass averaged to obtain the inlet condition for the next pass macros.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

6.1.3.4. Macro Heat Exchanger Group Connectivity

If the optional macro heat exchanger group is used, a single heat exchanger may be consist of multiple fluid zones. In this case, the auxiliary fluid is assumed to flow through these zones in parallel. Thus, after taking into account any auxiliary stream effects, the auxiliary fluid inlet mass flow rate is automatically apportioned to each zone in the group as follows:

$$\dot{m}_i = \left(\frac{\sum_k V_{i,k}}{\sum_i \sum_k V_{i,k}} \right) \dot{m} \tag{6.18}$$

where $\dot{m}_i$ is the total auxiliary mass flow rate for the heat exchanger group. $V_{i,k}$ refers to the volume of the k^{th} finite volume cell within the i^{th} fluid zone. Within each zone, the auxiliary fluid flows through each macro in series as usual.

At the outlet end of the group, the parallel auxiliary fluid streams through the individual zones are recombined, and the outlet auxiliary fluid enthalpy is calculated on a mass-averaged basis:

$$\overline{h} = \left(\frac{\sum_i \dot{m}_i h_i}{\sum_i \dot{m}_i} \right) \tag{6.19}$$

With user-defined functions, the **simple-effectiveness-model** allows you to simulate two-phase auxiliary fluid flows and other complex auxiliary fluid enthalpy relationships of the form

$$h = h(T, p, x) \tag{6.20}$$

where p is the absolute pressure and x is the quality (mass fraction of vapor) of a two-phase vapor-liquid mixture. When pressure-dependent auxiliary fluid properties are used, the mean pressure within each macro is calculated and passed to the user-defined function as

$$\overline{p}_j = p_{in} + \left(j + \frac{1}{2} \right) \frac{\Delta p}{N} \tag{6.21}$$

where

j = macro row index

p_{in} = inlet auxiliary fluid pressure

Δp = overall pressure drop across a heat exchanger group

N = number of rows per pass × number of passes.

To learn how to use the macro heat exchanger models, refer to Using the Ungrouped Macro Heat Exchanger Model and Using the Grouped Macro Heat Exchanger Model in the User's Guide.

6.2. The Dual Cell Model

The following sections contain information about the theory behind the dual cell heat exchanger model:

6.2.1. Overview of the Dual Cell Model

6.2.2. Restrictions of the Dual Cell Model

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

6.2.3. Dual Cell Model Theory

6.2.1. Overview of the Dual Cell Model

The Macro Model is quite suitable for thin rectangular heat exchanger cores, where the pass-to-pass is perpendicular to the primary flow direction and the auxiliary flow is uniform (so it can be treated as a 1D flow). Moreover, the mesh should be uniform and structured. However, many practical heat exchangers have a non-rectangular core and the auxiliary fluid, before reaching the core, may pass through arbitrary shaped inlet tanks, which make them highly non-uniform. It is quite possible that due to the complex shape of the core and or ease of meshing, the structured mesh may not be the obvious choice. These shortcomings of the macro model can be easily overcome by using the dual cell heat exchanger model. This model allows the solution of both the primary and auxiliary flow on separate co-located meshes and couples the two flows only through heat transfer at the heat exchanger core.

6.2.2. Restrictions of the Dual Cell Model

The following restrictions exist for the dual cell heat exchanger models:

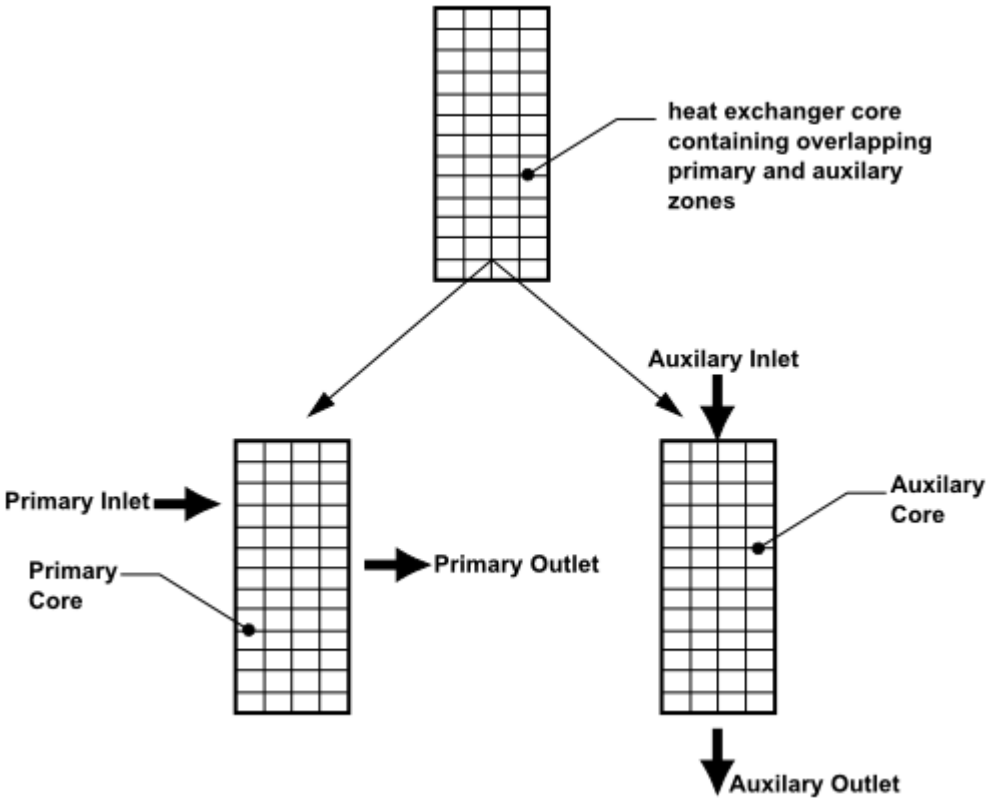
- The **simple-effectiveness-model** is not available.
- The heat exchanger performance data must be in the form of heat transfer rates or number of transfer units (NTU) for auxiliary / primary fluid flow rate combinations, rather than a velocity vs. effectiveness curve.
- In the case of a heat exchanger in which the primary and auxiliary meshes are not identical, heat transfer may be non-conservative (that is, the heat lost by the hot fluid may not equal the heat gained by the cold fluid). To minimize the difference in heat transfer, the topology and size of the primary and auxiliary cells should be as similar as possible, with the ideal being one-to-one cell conductivity.

6.2.3. Dual Cell Model Theory

The dual cell heat exchanger consists of two porous fluid zones, namely a primary zone and an auxiliary zone. The two zones are solved simultaneously and are coupled only through heat transfer. The common region in each zone, where heat transfer takes place, represents the heat exchanger core. The cores for both primary and auxiliary zones occupy the same physical space, as shown in Figure 6.3: Core with Matching Quad Meshes for Primary and Auxiliary Zones in a Crossflow Pattern (p. 233). The cells in the two cores should overlap completely in the physical space to ensure conservative heat transfer. Heat transfer occurs between cells in close proximity based on the cell centroid. That is, a primary zone cell exchanges heat with one, and only one, auxiliary zone cell and vice versa. Therefore, if one of the core (say primary) mesh is too coarse or fine relative to the other core (say auxiliary), conservation of heat transfer is not ensured. Heat transfer calculations in the dual cell model are based on the NTU method, regardless of whether you provide the raw heat rejection data or NTU data.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Figure 6.3: Core with Matching Quad Meshes for Primary and Auxiliary Zones in a Crossflow Pattern



6.2.3.1. NTU Relations

When you provide raw heat rejection data, the NTU values are calculated using Equation 6.9 (p. 229) and the **Effectiveness-NTU Relation** specified in the **Performance Data** tab of the **Set Dual Cell Heat Exchanger** dialog box. In a crossflow pattern, the NTU values are calculated as in Equation 6.10 (p. 229). The equation is solved iteratively using the Newton-Raphson. For parallel flow, the NTU value is calculated as follows:

$$NTU=\frac{-ln\left(1-\varepsilon-\varepsilon C_r\right)}{\left(C_r+1\right)} \tag{6.22}$$

and for counter flow, the following equation is used:

$$NTU_{C_r=1}=\frac{\varepsilon}{1-\varepsilon} \tag{6.23}$$

Otherwise,

$$NTU=\frac{1}{\left(C_r-1\right)}ln\left[\frac{\left(1-\varepsilon\right)}{\left(1-\varepsilon C_r\right)}\right] \tag{6.24}$$

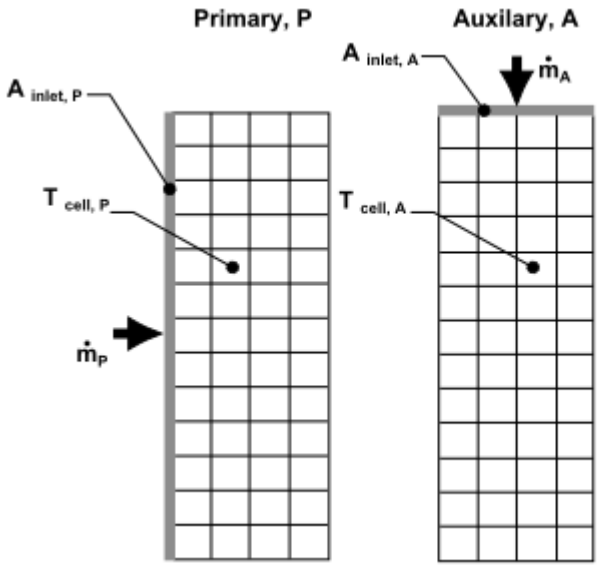
where C_r is the heat capacity ratio and ε is the effectiveness.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

6.2.3.2. Heat Rejection

Heat rejection is computed for each cell in the two cores (primary and auxiliary) and added as a source term to the energy equation for the respective flows. This is illustrated in Figure 6.4: Core with Primary and Auxiliary Zones with Overlap of Cells (p. 234).

Figure 6.4: Core with Primary and Auxiliary Zones with Overlap of Cells



$$\dot{m}_{scaled,A}=\rho_{cell,A}\vec{V}_{cell,A}A_{inlet,A} \tag{6.25}$$

$$\dot{m}_{scaled,P}=\rho_{cell,P}\vec{V}_{cell,P}A_{inlet,P} \tag{6.26}$$

$$C_{min,scaled}=min\left[\left(C_{p,cell}\dot{m}_{scaled}\right)\Big|_{P},\left(C_{p,cell}\dot{m}_{scaled}\right)\Big|_{A}\right] \tag{6.27}$$

$$NTU_{scaled}=\left[NTU_{full}\left(\dot{m}_{scaled,P},\dot{m}_{scaled,A}\right)\right]_{bilinearinterpolation} \tag{6.28}$$

$$\left(UA\right)_{scaled}=NTU_{scaled}C_{min,scaled} \tag{6.29}$$

$$q_{cell}=\frac{\left(UA\right)_{scaled}\left(T_{cell,A}-T_{cell,P}\right)V_{cell}}{V_{core}} \tag{6.30}$$

where

$T_{cell,A}$ = auxiliary cell temperature

$T_{cell,P}$ = primary cell temperature

UA = the product of the overall heat transfer coefficient (U) and the total heat transfer area (A)

$C_{min,scaled}$ = scaled minimum heat capacity rate

V_{cell} = volume of cell

V_{core} = volume of the heat exchanger core

You can supply the NTU values, or it can be calculated using the supplied raw data and the effectiveness-NTU relation that you specify.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

To learn how to use the dual cell heat exchanges model, refer to Using the Dual Cell Heat Exchanger Model in the User's Guide.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information
of ANSYS, Inc. and its subsidiaries and affiliates.

Chapter 7: Modelling with Finite-Rate Chemistry

This section describes the theory behind finite-rate chemistry used in Ansys Fluent:

7.1. Species Transport and Finite-Rate Chemistry

7.2. Composition PDF Transport

7.3. Chemistry Acceleration

7.1. Species Transport and Finite-Rate Chemistry

Ansys Fluent can model the mixing and transport of chemical species by solving conservation equations describing convection, diffusion, and reaction sources for each component species. Multiple simultaneous chemical reactions can be modeled, with reactions occurring in the fluid phase (volumetric reactions) and/or on wall or particle surfaces, and in the porous region. Species transport modeling capabilities, both with and without reactions, are described in this section.

Note that you may also want to consider modeling your turbulent reacting flame using the mixture fraction approach (for non-premixed systems, described in Non-Premixed Combustion (p. 295)), the reaction progress variable approach (for premixed systems, described in Premixed Combustion (p. 332)), the partially premixed approach (described in Partially Premixed Combustion (p. 343)), or the composition PDF Transport approach (described in Composition PDF Transport (p. 276)). Modeling multiphase species transport with finite-rate chemistry can be found in Multiphase Flows (p. 595).

Information is divided into the following sections:

7.1.1. Volumetric Reactions

7.1.2. Wall Surface Reactions and Chemical Vapor Deposition

7.1.3. Particle Reactions

7.1.4. Electrochemical Reactions

7.1.5. Reacting Channel Model

7.1.6. Reactor Network Model

For more information about using these models in Ansys Fluent, see Modeling Species Transport and Finite-Rate Chemistry in the User's Guide.

7.1.1. Volumetric Reactions

Theoretical information about species transport and finite-rate chemistry as related to volumetric reactions is presented in this section. This information is organized in the following sections:

7.1.1.1. Species Transport Equations

7.1.1.2. The Generalized Finite-Rate Formulation for Reaction Modeling

7.1.1.3. Finite-Rate Chemistry with the Two-Temperature Model

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

For more information about using species transport and finite-rate chemistry as related to volumetric reactions, see Volumetric Reactions in the User's Guide.

7.1.1.1. Species Transport Equations

When you choose to solve conservation equations for chemical species, Ansys Fluent predicts the local mass fraction of each species, Y_i, through the solution of a convection-diffusion equation for the i^{th} species. This conservation equation takes the following general form:

$$\frac{\partial}{\partial t}\left(\rho Y_i\right)+\nabla\cdot\left(\rho\vec{v}Y_i\right)=-\nabla\cdot\vec{J}_i+R_i+S_i \tag{7.1}$$

ρ = mixture density

$\vec{v}$ = velocity of the diffusing species

R_i= net rate of production of species i by chemical reaction (described later in this section)

S_i= rate of creation by addition from the dispersed phase plus any user-defined sources

An equation of this form will be solved for $N-1$ species where N is the total number of fluid phase chemical species present in the system. Since the mass fraction of the species must sum to unity, the N^{th} mass fraction is determined as one minus the sum of the $N-1$ solved mass fractions. To minimize numerical error, the N^{th} species should be selected as that species with the overall largest mass fraction, such as N_2 when the oxidizer is air.

7.1.1.1.1. Mass Diffusion in Laminar Flows

In Equation 7.1 (p. 238), $\vec{J}_i$ is the diffusion flux of species i, which arises due to gradients of concentration and temperature. By default, Ansys Fluent uses the dilute approximation (also called Fick's law) to model mass diffusion due to concentration gradients, under which the diffusion flux can be written as

$$\vec{J}_i=-\rho D_{i,m}\nabla Y_i-D_{T,i}\frac{\nabla T}{T} \tag{7.2}$$

where

$D_{i,m}$ = mass diffusion coefficient for species i in the mixture

Y_i = mass fraction of species i

$D_{T,i}$ = thermal (Soret) diffusion coefficient (see Thermal Diffusion Coefficients (p. 242))

T = temperature

For certain laminar flows, the dilute approximation may not be acceptable, and full multicomponent diffusion is required. In such cases, the Maxwell-Stefan equations can be solved; see Full Multicomponent Diffusion (p. 239) in the User's Guide for details.

Equation 7.2 (p. 238) is strictly valid when the mixture composition is not changing, or when $D_{i,m}$ is independent of composition. This is an acceptable approximation in dilute mixtures when $Y_i << 1$, for all i except the carrier gas. Ansys Fluent can also compute the transport of non-dilute mixtures in laminar flows by treating such mixtures as multicomponent system. Within Ansys Fluent, $D_{i,m}$ can be specified in a variety of ways, including by specifying $\mathcal{D}_{ij}$, the binary mass

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

diffusion coefficient of species i in species j. $\mathcal{D}_{ij}$ is not used directly, however; instead, the diffusion coefficient in the mixture, $D_{i,m}$, is computed as [298] (p. 1070):

$$D_{i,m}=\varepsilon\left(\sum_{j,\,j\neq i}\left(\frac{X_j}{\mathcal{D}_{ij}}\right)+\frac{X_i}{(1-Y_i)}\sum_{j,\,j\neq i}\left(\frac{Y_j}{\mathcal{D}_{ij}}\right)\right)^{-1} \tag{7.3}$$

where X_i and Y_i are the mole fraction and mass fraction of species i, respectively, and ε is the porosity. You can input $D_{i,m}$ or $\mathcal{D}_{ij}$ for each chemical species, as described in Mass Diffusion Coefficient Inputs in the *Fluent User's Guide*.

7.1.1.1.2. Mass Diffusion in Turbulent Flows

In turbulent flows, Ansys Fluent computes the mass diffusion in the following form:

$$\vec{J}_i=-\left(\rho D_{i,m}+\frac{\mu_t}{Sc_t}\right)\nabla Y_i-D_{T,i}\frac{\nabla T}{T} \tag{7.4}$$

where μ_t is the turbulent viscosity, and Sc_t is the turbulent Schmidt number calculated as:

$$Sc_t=\frac{\mu_t}{\rho D_t} \tag{7.5}$$

where D_t is the turbulent diffusivity. The default value of Sc_t is 0.7. Note that turbulent diffusion generally overwhelms laminar diffusion, and the specification of detailed laminar diffusion properties in turbulent flows is generally not necessary.

In turbulent flows, the total diffusion is the sum of the laminar diffusion $D_{i,m}$ given by Equation 7.3 (p. 239) and the turbulent diffusion given by Equation 7.5 (p. 239)

7.1.1.1.3. Full Multicomponent Diffusion

A careful treatment of chemical species diffusion in the species transport and energy equations is important when details of the molecular transport processes are significant (for example, in diffusion-dominated laminar flows). As one of the laminar-flow diffusion models, Ansys Fluent has the ability to model full multicomponent species transport.

7.1.1.1.3.1. General Theory

For multicomponent systems, it is not possible, in general, to derive relations for the diffusion fluxes containing the gradient of only one component (as described in Mass Diffusion in Laminar Flows (p. 238)). Here, the Maxwell-Stefan equations will be used to obtain the diffusive mass flux. This will lead to the definition of generalized Fick's law diffusion coefficients [646] (p. 1090). This method is preferred over computing the multicomponent diffusion coefficients since their evaluation requires the computation of N^2 co-factor determinants of size $(N-1)\times(N-1)$, and one determinant of size $N\times N$ [633] (p. 1089), where N is the number of chemical species.

7.1.1.1.3.2. Maxwell-Stefan Equations

From Merk [433] (p. 1078), the Maxwell-Stefan equations can be written as

$$\sum_{\substack{j=1\\ j\neq i}}^{N}\frac{X_iX_j}{\mathcal{D}_{ij}}\left(\vec{V}_j-\vec{V}_i\right)=\vec{d}_i-\frac{\nabla T}{T}\sum_{\substack{j=1\\ j\neq i}}^{N}\frac{X_iX_j}{\mathcal{D}_{ij}}\left(\frac{D_{T,j}}{\rho_j}-\frac{D_{T,i}}{\rho_i}\right) \tag{7.6}$$

where,

X = the mole fraction

$\vec{V}$ = the diffusion velocity

$\mathcal{D}_{ij}$ = the binary mass diffusion coefficient of species i in species j

D_T = the thermal diffusion coefficient.

For an ideal gas the Maxwell diffusion coefficients are equal to the binary diffusion coefficients. If the external force is assumed to be the same on all species and that pressure diffusion is negligible, then $\vec{d}_i = \nabla X_i$. Since the diffusive mass flux vector is $\vec{J}_i = \rho_i \vec{V}_i$, the above equation can be written as

$$\sum_{\substack{j=1 \\ j \neq i}}^{N} \frac{X_i X_j}{\mathcal{D}_{ij}} \left(\frac{\vec{J}_j}{\rho_j} - \frac{\vec{J}_i}{\rho_i} \right) = \nabla X_i - \frac{\nabla T}{T} \sum_{\substack{j=1 \\ j \neq i}}^{N} \frac{X_i X_j}{\mathcal{D}_{ij}} \left(\frac{D_{T,j}}{\rho_j} - \frac{D_{T,i}}{\rho_i} \right) \tag{7.7}$$

After some mathematical manipulations, the diffusive mass flux vector, $\vec{J}_i$, can be obtained from

$$\vec{J}_i = -\sum_{j=1}^{N-1} \rho D_{ij} \nabla Y_j - D_{T,i} \frac{\nabla T}{T} \tag{7.8}$$

where Y_j is the mass fraction of species j. Other terms are defined as follows:

$$\begin{aligned}
D_{ij} &= [D] = [A]^{-1}[B] \\
M_{w,m} &= \left(\sum_{i=1}^{N} \frac{Y_i}{M_{w,i}} \right) \\
A_{ii} &= -\left(\frac{X_i}{\mathcal{D}_{iN}} \frac{M_{w,m}}{M_{w,N}} + \sum_{\substack{j=1 \\ j \neq i}}^{N} \frac{X_j}{\mathcal{D}_{ij}} \frac{M_{w,m}}{M_{w,i}} \right) \\
A_{ij} &= X_i \left(\frac{1}{\mathcal{D}_{ij}} \frac{M_{w,m}}{M_{w,j}} - \frac{1}{\mathcal{D}_{iN}} \frac{M_{w,m}}{M_{w,N}} \right) \\
B_{ii} &= -\left(X_i \frac{M_{w,m}}{M_{w,N}} + (1 - X_i) \frac{M_{w,m}}{M_{w,i}} \right) \\
B_{ij} &= X_i \left(\frac{M_{w,m}}{M_{w,j}} - \frac{M_{w,m}}{M_{w,N}} \right)
\end{aligned} \tag{7.9}$$

where,

$[A]$ and $[B]$ = $(N-1) \times (N-1)$ matrices

$[D]$ = $(N-1) \times (N-1)$ matrix of the generalized Fick's law diffusion coefficients D_{ij} [646] (p. 1090)

M_w = the molecular weight

m = a subscript used to define a mean molecular weight in the mixture

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

7.1.1.1.4. Kinetic Theory Parameters for Diffusion Coefficients

The solver will use a modification of the Chapman-Enskog formula [418] (p. 1077) to compute the diffusion coefficient $\mathcal{D}_{ij}$ (cm^2/s) using kinetic theory:

$$\mathcal{D}_{ij}=0.00186\frac{\left[T^3\left(\frac{1}{M_{w,i}}+\frac{1}{M_{w,j}}\right)\right]^{1/2}}{p_{abs}\sigma_{ij}^2\Omega_D} \tag{7.10}$$

where,

M_w = molar mass (g/mol)

T = temperature (K)

p_{abs} = absolute pressure (atm)

Ω_D = non-dimensional diffusion collision integral, which is a measure of the interaction of the molecules in the system

Ω_D is a function of the dimensionless temperature T_D^*, which is calculated as:

$$T_D^*=\frac{T}{\left(\varepsilon/k_B\right)_{ij}} \tag{7.11}$$

where T is the temperature (in Kelvin).

k_B is the Boltzmann constant, which is defined as the gas constant, R, divided by Avogadro's number. $\left(\varepsilon/k_B\right)_{ij}$ in Kelvin for the mixture is the *geometric* average:

$$\left(\varepsilon/k_B\right)_{ij}=\sqrt{\left(\varepsilon/k_B\right)_i\left(\varepsilon/k_B\right)_j} \tag{7.12}$$

The relation between Ω_D and T_D^* is obtained from the tabulation values given in [247] (p. 1067), which is approximated in Ansys Fluent as:

$$\Omega_D=\frac{1}{T_D^{*0.145}}+\frac{1}{\left(T_D^*+0.5\right)^2} \tag{7.13}$$

For a binary mixture, σ_{ij} (in angstroms, where 1 angstrom = $1.0*10^{-10}$ m) is calculated as the *arithmetic* average of the individual σs:

$$\sigma_{ij}=\frac{1}{2}\left(\sigma_i+\sigma_j\right) \tag{7.14}$$

7.1.1.1.5. Unity Lewis Number

The laminar mass diffusivity of all species is calculated as:

$$D_{i,m}=\frac{k}{\rho c_p}$$

where $D_{i,m}$ denotes the mass diffusivity of species i in the mixture, k is the thermal conductivity, ρ is the mixture density, and c_p is the mixture specific heat.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

7.1.1.1.6. Anisotropic Species Diffusion

Anisotropic species diffusion in porous media is modeled as follows:

- For Fickian diffusion, the mass flux vector of species i is modeled as,

$$J_i = -\varepsilon\left(\rho D_{i,m} + \frac{\mu_t}{Sc_t}\right)\mathbf{K}\nabla Y_i - \varepsilon D_{T,i}\frac{\nabla T}{T} \tag{7.15}$$

  where ε is the porosity, $\mathbf{K}$ is the anisotropic diffusion matrix in the porous zone, and the remaining nomenclature is the same as in Equation 7.4 (p. 239).

- The full multi-component (Maxwell-Stefan) anisotropic diffusion flux vector is calculated similar to Equation 7.8 (p. 240) as,

$$J_i = -\varepsilon\sum_{j=1}^{N-1}\rho D_{ij}\mathbf{K}\nabla Y_i - \varepsilon D_{T,i}\frac{\nabla T}{T} \tag{7.16}$$

7.1.1.1.7. Thermal Diffusion Coefficients

The thermal diffusion coefficients can be defined as constants, polynomial functions, user-defined functions, or using the following empirically-based composition-dependent expression derived from [325] (p. 1072):

$$D_{T,i} = -2.59\times10^{-7}T^{0.659}\left[\frac{M_{w,i}^{0.511}X_i}{\sum_{i=1}^{N}M_{w,i}^{0.511}X_i} - Y_i\right]\cdot\left[\frac{\sum_{i=1}^{N}M_{w,i}^{0.511}X_i}{\sum_{i=1}^{N}M_{w,i}^{0.489}X_i}\right] \tag{7.17}$$

This form of the Soret diffusion coefficient will cause heavy molecules to diffuse less rapidly, and light molecules to diffuse more rapidly, towards heated surfaces.

7.1.1.1.8. Treatment of Species Transport in the Energy Equation

For many multicomponent mixing flows, the transport of enthalpy due to species diffusion

$$\nabla\cdot\left[\sum_{i=1}^{n}h_i\vec{J}_i\right]$$

can have a significant effect on the enthalpy field and should not be neglected. In particular, when the Lewis number

$$Le_i = \frac{k}{\rho c_p D_{i,m}} \tag{7.18}$$

for any species is far from unity, neglecting this term can lead to significant errors. Ansys Fluent will include this term by default. In Equation 7.18 (p. 242), k is the thermal conductivity.

7.1.1.1.9. Diffusion at Inlets

For the pressure-based solver in Ansys Fluent, the net transport of species at inlets consists of both convection and diffusion components. For the density-based solvers, only the convection component is included. The convection component is fixed by the user-specified inlet species

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

mass fraction. The diffusion component, however, depends on the gradient of the computed species field at the inlet. Thus the diffusion component (and therefore the net inlet transport) is not specified a priori. For information about specifying the net inlet transport of species, see Defining Cell Zone and Boundary Conditions for Species in the User's Guide.

7.1.1.2. The Generalized Finite-Rate Formulation for Reaction Modeling

The reaction rates that appear as source terms in Equation 7.1 (p. 238) are computed in Ansys Fluent, for turbulent flows, by one of three approaches:

- Direct use of finite-rate kinetics: The effect of turbulent fluctuations on kinetics rates are neglected, and reaction rates are determined by general finite-rate chemistry directly.
- Eddy-dissipation model: Reaction rates are assumed to be controlled by the turbulence, ignoring the effect of chemistry timescales, which avoids expensive Arrhenius chemical kinetic calculations. The model is computationally cheap, but, for realistic results, only one or two step heat-release mechanisms should be used. This approach should be used only when the chemistry timescales of interest are known to be fast relative to the turbulence timescales throughout the domain.
- Eddy-dissipation-concept (EDC) model: Detailed chemical kinetics can be incorporated in turbulent flames, considering timescales of both turbulence and kinetics. Note that detailed chemical kinetic calculations can be computationally expensive.

The generalized finite-rate formulation is suitable for a wide range of applications including laminar or turbulent reaction systems, and combustion systems with premixed, non-premixed, or partially-premixed flames.

7.1.1.2.1. Direct Use of Finite-Rate Kinetics (no TCI)

When no turbulence-chemistry interaction (TCI) model is used, finite-rate kinetics are incorporated by computing the chemical source terms using general reaction-rate expressions, without attempting to account explicitly for the effects of turbulent fluctuations on the source-term calculations. This approach is recommended for laminar flows, where the formulation is exact, or for turbulent flows using complex chemistry where either the turbulence time-scales are expected to be fast relative to the chemistry time scales or where the chemistry is sufficiently complex that the chemistry timescales of importance are highly disparate.

The net source of chemical species i due to reaction is computed as the sum of the reaction sources over the N_R reactions that the species participate in. For reactions in a porous medium, the net source due to reactions is obtained by multiplying the reaction rates with porosity:

$$R_i = M_{w,i} \sum_{N_R}^{r=1} \varepsilon \hat{R}_{i,r} \tag{7.19}$$

where $M_{w,i}$ is the molecular weight of species i, ε is the porosity, and $\hat{R}_{i,r}$ is the molar rate of creation/destruction of species i in reaction r. Reaction may occur in the continuous phase at wall surfaces.

Consider the r^{th} reaction written in general form as follows:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$\sum_{i=1}^{N} \nu'_{i,r} \mathcal{M}_i \underset{k_{b,r}}{\overset{k_{f,r}}{\rightleftarrows}} \sum_{i=1}^{N} \nu''_{i,r} \mathcal{M}_i \tag{7.20}$$

where,

N = number of chemical species in the system

$\nu'_{i,r}$ = stoichiometric coefficient for reactant i in reaction r

$\nu''_{i,r}$ = stoichiometric coefficient for product i in reaction r

$\mathcal{M}_i$ = symbol denoting species i

$k_{f,r}$ = forward rate constant for reaction r

$k_{b,r}$ = backward rate constant for reaction r

Equation 7.20 (p. 244) is valid for both reversible and irreversible reactions. For irreversible reactions, the backward rate constant, $k_{b,r}$, is zero.

The summations in Equation 7.20 (p. 244) are for all chemical species in the system, but only species that appear as reactants or products will have nonzero stoichiometric coefficients.

The molar rate of creation/destruction of species i in reaction r is given by

$$\hat{R}_{i,r} = \Gamma \left(\nu''_{i,r} - \nu'_{i,r} \right) \left(k_{f,r} \prod_{j=1}^{N} \left[C_{j,r} \right]^{\eta'_{j,r}} - k_{b,r} \prod_{j=1}^{N} \left[C_{j,r} \right]^{\eta''_{j,r}} \right) \tag{7.21}$$

where,

$C_{j,r}$ = molar concentration of species j in reaction r $\left(\mathrm{kmol/m^3}\right)$

$\eta'_{j,r}$ = rate exponent for reactant species j in reaction r

$\eta''_{j,r}$ = rate exponent for product species j in reaction r

Note that the rate exponent for the reverse reaction part in Equation 7.21 (p. 244) is the product species stoichiometric coefficient ($\nu''_{j,r}$).

For information on more general reaction mechanisms in CHEMKIN format, see Getting Started with Ansys Chemkin. For information about inputting the stoichiometric coefficients and rate exponents within the Ansys Fluent user interface, for both global forward (irreversible) reactions and elementary (reversible) reactions, see Inputs for Reaction Definition in the *Fluent User's Guide*. The information below refers to mechanisms manually entered in Ansys Fluent.

Γ represents the net effect of third bodies on the reaction rate. This term is given by

$$\Gamma = \sum_{j}^{N} \gamma_{j,r} C_j \tag{7.22}$$

where $\gamma_{j,r}$ is the third-body efficiency of the j^{th} species in the r^{th} reaction. By default, Ansys Fluent does not include third-body effects in the reaction rate calculation. You can, however, opt to include the effect of third-body efficiencies if you have data for them.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

The forward rate constant for reaction r, $k_{f,r}$, is computed using the Arrhenius expression

$$k_{f,r}=A_r T^{\beta_r} e^{-E_r/RT} \tag{7.23}$$

where,

A_r = pre-exponential factor (consistent units)

β_r = temperature exponent (dimensionless)

E_r = activation energy for the reaction (J/kmol)

R = universal gas constant (J/kmol-K)

You (or the database) will provide values for $\nu'_{i,r}$, $\nu''_{i,r}$, $\eta'_{j,r}$, $\eta''_{j,r}$, β_r, A_r, E_r, and, optionally, $\gamma_{j,r}$ during the problem definition in Ansys Fluent.

If the reaction is reversible, the backward rate constant for reaction r, $k_{b,r}$, is computed, by default, from the forward rate constant using the following relation:

$$k_{b,r}=\frac{k_{f,r}}{K_r} \tag{7.24}$$

where K_r is the equilibrium constant for the r^{th} reaction, computed from

$$K_r=\exp\left(\frac{\Delta S_r}{R}-\frac{\Delta H_r}{RT}\right)\left(\frac{p_{\text{atm}}}{RT}\right)^{\sum_{i=1}^{N}(\nu''_{i,r}-\nu'_{i,r})} \tag{7.25}$$

where p_{atm} denotes atmospheric pressure (101325 Pa). The term within the exponential function represents the change in Gibbs free energy, and its components are computed as follows:

$$\frac{\Delta S_r}{R}=\sum_{i=1}^{N}\left(\nu''_{i,r}-\nu'_{i,r}\right)\frac{S_i}{R} \tag{7.26}$$

$$\frac{\Delta H_r}{RT}=\sum_{i=1}^{N}\left(\nu''_{i,r}-\nu'_{i,r}\right)\frac{h_i}{RT} \tag{7.27}$$

where S_i and h_i are the entropy and enthalpy of the i^{th} species evaluated at temperature T and atmospheric pressure. These values are specified in Ansys Fluent as properties of the mixture material.

Ansys Fluent also provides the option to explicitly specify the reversible reaction rate parameters (pre-exponential factor, temperature exponent, and activation energy for the reaction) if desired. In this case, the backward rate constant of the reversible reaction r is computed using the relationship similar to Equation 7.23 (p. 245):

$$k_{b,r}=A_{b,r} T^{\beta_{b,r}} e^{-E_{b,r}/RT} \tag{7.28}$$

where,

$A_{b,r}$ = backward reaction pre-exponential factor (consistent units)

$\beta_{b,r}$ = backward reaction temperature exponent (dimensionless)

$E_{b,r}$ = backward reaction activation energy for the reaction (J/kmol)

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

For more information about specifying the backward reaction rate parameters, see Inputs for Reaction Definition in the *Fluent User's Guide*.

7.1.1.2.2. Pressure-Dependent Reactions

General formulations for pressure-dependent reactions are available through use of CHEMKIN imported mechanisms and detailed descriptions of these formulations can be found in Getting Started with Ansys Chemkin. If you choose to manually enter reaction expressions through the Ansys Fluent interface, Ansys Fluent can use one of three methods to represent the rate expression in pressure-dependent (or pressure fall-off) reactions. These are a subset of the formulations available through use of CHEMKIN imported mechanisms. A "fall-off" reaction is one in which the temperature and pressure are such that the reaction occurs between Arrhenius high-pressure and low-pressure limits, and therefore is no longer solely dependent on temperature.

There are three methods of representing the rate expressions in this fall-off region. The simplest one is the Lindemann [374] (p. 1074) form. There are also two other related methods, the Troe method [206] (p. 1064) and the SRI method [631] (p. 1089), that provide a more accurate description of the fall-off region.

Arrhenius rate parameters are required for both the high- and low-pressure limits. The rate coefficients for these two limits are then blended to produce a smooth pressure-dependent rate expression. In Arrhenius form, the parameters for the high-pressure limit (k) and the low-pressure limit (k_{low}) are as follows:

$$k = AT^{\beta}e^{-E/RT} \tag{7.29}$$

$$k_{low} = A_{low}T^{\beta_{low}}e^{-E_{low}/RT} \tag{7.30}$$

The net rate constant at any pressure is then taken to be

$$k_{net}=k\left(\frac{p_r}{1+p_r}\right)F \tag{7.31}$$

where p_r is defined as

$$p_r=\frac{k_{low}[M]}{k} \tag{7.32}$$

and $[M]$ is the concentration of the bath gas, which can include third-body efficiencies. If the function F in Equation 7.31 (p. 246) is unity, then this is the Lindemann form. Ansys Fluent provides two other forms to describe F, namely the Troe method and the SRI method.

In the Troe method, F is given by

$$\log F=\left\{1+\left[\frac{\log p_r+c}{n-d\left(\log p_r+c\right)}\right]^2\right\}^{-1}\log F_{cent} \tag{7.33}$$

where

$$\begin{aligned} c &= -0.4-0.67\log F_{cent} \\ n &= 0.75-1.27\log F_{cent} \\ d &= 0.14 \end{aligned} \tag{7.34}$$

and

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$F_{cent}=\left(1-\alpha\right)e^{-T/T_1}+\alpha e^{-T/T_2}+e^{-T_3/T} \tag{7.35}$$

The parameters α, T_3, T_2, and T_1 are specified as inputs.

In the SRI method, the blending function F is approximated as

$$F=d\left[a\exp\left(\frac{-b}{T}\right)+\exp\left(\frac{-T}{c}\right)\right]^{X}T^{e} \tag{7.36}$$

where

$$X=\frac{1}{1+\log^{2}p_r} \tag{7.37}$$

In addition to the three Arrhenius parameters for the low-pressure limit (k_{low}) expression, you must also supply the parameters a, b, c, d, and e in the F expression.

For Chemically Activated Bimolecular pressure dependent reactions, the net rate constant at any pressure is

$$k_{net}=k_{low}\left(\frac{1}{1+p_r}\right)F \tag{7.38}$$

Important:

Chemical kinetic mechanisms usually contain a wide range of time scales and form a set of highly nonlinear, stiff coupled equations. For solution procedure guidelines, see Solution Procedures for Chemical Mixing and Finite-Rate Chemistry in the User's Guide. Also, if you have a chemical mechanism in CHEMKIN [299] (p. 1070) format, you can import this mechanism into Ansys Fluent. See Importing a Volumetric Kinetic Mechanism in CHEMKIN Format in the User's Guide.

7.1.1.2.3. The Eddy-Dissipation Model

Under some combustion conditions, fuels burn quickly and the overall rate of reaction is controlled by turbulent mixing. In high-temperature non-premixed flames, for example, turbulence slowly convects/mixes fuel and oxidizer into the reaction zones where they burn quickly. In certain premixed flames, the turbulence slowly convects/mixes cold reactants and hot products into the reaction zones, where reaction occurs rapidly. In such cases, one approximation is to assume the combustion is mixing-limited, allowing neglect of the complex chemical kinetic rates and instead assuming instantaneous burn upon mixing.

For mixed-is-burned approximation, Ansys Fluent provides a turbulence-chemistry interaction model, based on the work of Magnussen and Hjertager [407] (p. 1076), called the eddy-dissipation model. With this model, the net rate of production of species i due to reaction r, $R_{i,r}$, is given by the smaller (that is, limiting value) of the two expressions below:

$$R_{i,r}=(\nu''_{i,r}-\nu'_{i,r})M_{w,i}A\rho\frac{\varepsilon}{k}\min_{\mathfrak{R}}\left(\frac{Y_{\mathfrak{R}}}{\nu'_{\mathfrak{R},r}M_{w,\mathfrak{R}}}\right) \tag{7.39}$$

$$R_{i,r}=(\nu''_{i,r}-\nu'_{i,r})M_{w,i}AB\rho\frac{\varepsilon}{k}\frac{\sum_{P}Y_P}{\sum_{j}^{N}\nu''_{j,r}M_{w,j}} \tag{7.40}$$

where,

Y_P = the mass fraction of any product species, P

$Y_{\mathfrak{R}}$ = the mass fraction of a particular reactant, $\mathfrak{R}$

A = an empirical constant equal to 4.0

B = an empirical constant equal to 0.5

In Equation 7.39 (p. 247) and Equation 7.40 (p. 247), the chemical reaction rate is governed by the large-eddy mixing time scale, k/ε, as in the eddy-breakup model of Spalding [621] (p. 1089). Combustion proceeds whenever turbulence is present ($k/\varepsilon > 0$), and an ignition source is not required to initiate combustion. This is usually acceptable for non-premixed flames, but in premixed flames, the reactants will burn as soon as they enter the computational domain, upstream of the flame stabilizer. To remedy this, Ansys Fluent provides the finite-rate/eddy-dissipation model, where both the finite-rate reaction rates (Equation 7.21 (p. 244)), and eddy-dissipation (Equation 7.39 (p. 247) and Equation 7.40 (p. 247)) rates are calculated. The net reaction rate is taken as the minimum of these two rates. In practice, the finite-rate kinetics acts as a kinetic "switch", preventing reaction before the flame holder. Once the flame is ignited, the eddy-dissipation rate is generally smaller than the Arrhenius rate, and reactions are mixing-limited.

Important:

- Although Ansys Fluent allows multi-step reaction mechanisms (number of reactions >2) with the eddy-dissipation and finite-rate/eddy-dissipation models, these will likely produce incorrect solutions. The reason is that multi-step chemical mechanisms are typically based on Arrhenius rates, which differ for each reaction. In the eddy-dissipation model, every reaction has the same, turbulent rate, and therefore the model should be used only for one-step (reactant → product), or two-step (reactant → intermediate, intermediate → product) global reactions. The model cannot predict kinetically controlled species such as radicals. To incorporate multi-step chemical kinetic mechanisms in turbulent flows, use the EDC model (described in The Eddy-Dissipation Model for LES (p. 248)).
- The eddy-dissipation model requires products to initiate reaction (see Equation 7.40 (p. 247)). When you initialize the solution for steady flows, Ansys Fluent sets all species mass fractions to a maximum in the user-specified initial value and 0.01. This is usually sufficient to start the reaction. However, if you converge a mixing solution first, where all product mass fractions are zero, you may then have to patch products into the reaction zone to ignite the flame. For details, see Ignition in Steady-State Combustion Simulations in the User's Guide.

7.1.1.2.4. The Eddy-Dissipation Model for LES

When the LES turbulence model is used, the turbulent mixing rate, ε/k in Equation 7.39 (p. 247) and Equation 7.40 (p. 247), is replaced by the subgrid-scale mixing rate. This is calculated as

$$\tau_{sgs}^{-1} = \sqrt{2S_{ij}S_{ij}} \tag{7.41}$$

where,

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

τ_{sgs}^{-1} = subgrid-scale mixing rate (s^{-1})

$S_{ij} = \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right)$ = strain rate tensor (s^{-1})

7.1.1.2.5. The Eddy-Dissipation-Concept (EDC) Model

The eddy-dissipation-concept (EDC) model is an extension of the eddy-dissipation model that allows inclusion of detailed chemical mechanisms in turbulent flows [406] (p. 1076). It assumes that reactions occur within small turbulent structures, called the fine scales. These fine scales occupy a fraction of the computational cell, characterized by a volume fraction κ. Combustion at the fine scales is assumed to occur as a constant pressure reactor, with initial conditions taken as the current species and temperature in the cell. Reactions proceed over the time scale τ^*, governed by the reaction rates of Equation 7.21 (p. 244), and are integrated numerically using an ODE solver. The source term in the conservation equation for the mean species i, Equation 7.1 (p. 238), is modeled as

$$R_i = \rho\kappa\frac{Y_i^* - Y_i}{\tau^*} \tag{7.42}$$

where Y_i^* is the fine-scale species mass fraction after reacting over the time scale τ^*.

The EDC model can incorporate detailed chemical mechanisms into turbulent reacting flows. However, typical mechanisms are invariably stiff and their numerical integration is computationally costly. Hence, the model should be used only when the assumption of fast chemistry is invalid, such as modeling low-temperature or high-pressure combustion, the slow CO burnout in rapidly quenched flames, or the NO conversion in selective non-catalytic reduction (SNCR).

For guidelines on obtaining a solution using the EDC model, see Solution of Stiff Chemistry Systems in the User's Guide.

7.1.1.2.5.1. The Standard EDC Model

In the standard EDC model [217] (p. 1065), the volume fraction of the reactive fine scales is modeled as:

$$\kappa = \frac{\xi^{*2}}{1 - \xi^{*3}} \tag{7.43}$$

where the length scale ξ^* is computed as:

$$\xi^* = C_\xi\left(\frac{\nu\varepsilon}{k^2}\right)^{1/4} \tag{7.44}$$

where

C_ξ = length scale constant = 2.1377

ν = kinematic viscosity

k = turbulent kinetic energy

ε = rate of dissipation of turbulent kinetic energy

The integration time scale τ^* is modeled as:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$\tau^{*}=C_{\tau}\left(\frac{\nu}{\varepsilon}\right)^{1/2} \tag{7.45}$$

where C_{τ} is a time scale constant equal to 0.4082.

7.1.1.2.5.2. The Partially Stirred Reactor EDC Model

The standard EDC model (see The Standard EDC Model (p. 249) for details), has been successfully applied in modeling turbulence-chemistry coupling by incorporating detailed chemical mechanisms. However, the main shortcoming of the standard EDC model is that neither the length scale constant C_{ξ} nor the time scale constant C_{τ} depends on local flow and chemistry variables. Studies have shown limitations of the standard EDC model. For example, when applied to the simulation of Moderate and Intense Low-oxygen Dilution (MILD) systems (slow chemistry due to strong oxygen dilution), the EDC model gives a significant overestimation of temperature levels [359] (p. 1073), [360] (p. 1074).

Several authors have offered the generalization of the EDC model for applications to a wider flame regime. The Partially Stirred Reactor (PaSR) model [176] (p. 1063), [717] (p. 1094) has been applied to both mild and supersonic combustions. It is conceptually similar to the standard EDC model, but is characterized by a different definition of the reacting volume fraction and time scale:

$$\kappa=\frac{t_c}{t_c+t_{mix}}=\frac{1}{1+Da} \tag{7.46}$$

$$\tau^{*}=Min\left(t_c,\ t_{mix}\right) \tag{7.47}$$

In Equation 7.46 (p. 250), Da is defined as:

$$Da=\frac{t_{mix}}{t_c} \tag{7.48}$$

In Ansys Fluent, the chemistry time scale t_c is computed based upon the practice outlined in [166] (p. 1062):

$$t_c=\max\left(\rho y_i/\omega_i\right) \tag{7.49}$$

where ω_i are the reaction rates (in kg/m^3/s) of CH_4, H_2, O_2, H_2O, and CO_2 as these slower major species dominate the fine-scale time-scales and larger flame structure.

The mixing time scale t_{mix} is taken as a fraction of the integral time scale of turbulence $\frac{k}{\varepsilon}$:

$$t_{mix}=C_{mix}\frac{k}{\varepsilon} \tag{7.50}$$

where C_{mix} can be a constant value or a function of the local turbulent Reynolds number ($Re_T=\ \mu_t/\mu_l$):

$$C_{mix}=\left(\frac{C_{\mu}}{\mu_t/\mu_l}\right)^{(1-\alpha)/2}=\left(\frac{Re_T}{C_{\mu}}\right)^{(\alpha-1)/2} \tag{7.51}$$

where C_{μ} is the constant of the k-ε turbulence model.

When Equation 7.51 (p. 250) is used to compute C_{mix}, the model is referred to as the fractal method [176] (p. 1063). The exponent coefficient α is expressed as:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$\alpha=\frac{3\left(D\text{-}3\right)}{1+D} \tag{7.52}$$

where D is the fractal dimension. A value of D = 3 corresponds to the Kolmogorov time-scale, whereas a value of D = 5 results in the integral time.

7.1.1.2.6. The Thickened Flame Model

Premixed flames have typical laminar flame thicknesses of the order of a millimeter. Since the laminar flame propagation speed is determined by species diffusion, heat conduction and chemical reaction within the flame, sufficient grid resolution inside the flame is necessary to predict the correct laminar flame velocity. In contrast, the combustor dimensions are usually much larger than the laminar flame thickness and the flame cannot be affordably resolved, even with unstructured and solution-adaptive grids.

The premixed laminar flame speed, denoted S_l^0, is proportional to $\sqrt{DR}$ where D is a diffusivity and R is a reaction rate. The laminar flame thickness is proportional to D/S_l^0. Hence, the laminar flame can be artificially thickened, without altering the laminar flame speed, by increasing the diffusivity and decreasing the reaction rate proportionally. The thickened flame can then be feasibly resolved on a coarse mesh while still capturing the correct laminar flame speed.

The thickening factor F is calculated in Ansys Fluent as:

$$F=\frac{N\Delta}{\delta} \tag{7.53}$$

where Δ is the grid size, δ is the laminar flame thickness, and N is the user-specified number of points in the flame (default of 8). The grid-size, Δ, is determined as $V^{1/d}$ where V is the cell volume and d is the spatial dimension (2 or 3). The laminar flame thickness, δ, is a user-input, and may be specified as a constant, a User-Defined Function, or calculated as D/S_l^0, where D is the thermal diffusivity evaluated as $k/\rho c_p$ (k is the thermal conductivity, ρ is the density and c_p is the specific heat).

All species diffusion coefficients, as well as the thermal conductivity, are multiplied by the thickening factor, F, and all reaction rates are divided by F. However, away from the flame, these enhanced diffusivities can cause erroneous mixing and heat-transfer, so the flame is dynamically thickened only in a narrow band around the reaction front. This band is calculated by multiplying F with, a factor Ω calculated as

$$\Omega=\tanh\left(\beta\frac{\overline{|R|}}{\max(\overline{|R|})}\right) \tag{7.54}$$

In Equation 7.54 (p. 251), $\overline{|R|}$ is the spatially filtered absolute value of the reaction rate, and β is a constant with a default value of 10. The absolute reaction rate is filtered several times, and $\max(\overline{|R|})$ is the maximum value of $\overline{|R|}$ in the domain. Ω ranges from unity in the band around the flame to zero outside this band.

The Thickened Flame Model (TFM) [494] (p. 1082) is most often used with a single step chemical mechanism where the global reaction has been tuned to provide the correct laminar flame speed. The TFM can, in principal, be used with multi-step reaction mechanisms, however all composition

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

profiles should be adequately resolved within the flame. The TFM is available in both the pressure-based as well as density-based solvers.

While the TFM can be used to model laminar flames, its most common application is as an LES combustion model for turbulent premixed and partially-premixed flames. The turbulent flame speed of a premixed flame is determined principally by the flame wrinkling, which increases the flame surface area.

The thickening of the flame alters its interaction with the turbulence. This should be considered in the calculations of the species formation and destruction rates. As the flame thickens, the ability of turbulence to wrinkle the flame diminishes, which impacts the turbulent flame speed propagation. The decrease in the turbulence chemistry interaction is compensated by multiplying the reaction rates by the efficiency function E. In Ansys Fluent, the Charlette and Colin efficiency functions are available for modeling this phenomenon.

The Colin efficiency function is computed by:

$$E = \frac{1 + \alpha \Gamma_0 \frac{u'_{\Delta_e}}{s_l^0}}{1 + \alpha \Gamma_1 \frac{u'_{\Delta_e}}{s_l^0}} \tag{7.55}$$

where,

u'_{Δ_e} = sub grid scale turbulent velocity computed using the method proposed by Durand and Polifke [154] (p. 1061).

Δ_e = filter width that corresponds to the thickened flame, which is usually an order of magnitude higher than the physical grid length scale. The default value is taken 10 times the local cell grid spacing.

s_l^0 = laminar flame speed.

α = parameter computed by Equation 7.56 (p. 252).

Γ_0 and Γ_1 = are defined by Equation 7.58 (p. 253).

$$\alpha = \beta \frac{2\ln(2)}{3 c_{ms} \left[\xi \sqrt{Re_t} - 1 \right]} \tag{7.56}$$

where,

β = user-defined parameter of the order of unity.

ξ = correction parameter that varies from 0 to 1. The correction is based on the local turbulence conditions and the value of the cut-off length, which is the smallest length scale below which the eddies do not alter the flame. The cut-off length is computed as a function of the thickened flame thickness and the Reynolds number.

c_{ms} = is a constant with a value of 0.28.

The turbulent Reynolds number Re_t in Equation 7.56 (p. 252) is based on the integral length scales l_t and the RMS velocity fluctuations at integral length scales. The integral length scale is the length of the largest eddies present in the domain. It depends on the geometry, and a value of

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

1/4 to 1/2 of a characteristics dimension (such as a burner diameter, an inlet diameter, or a size of a bluff body) is typically used. The RMS velocity fluctuations at the integral length scale is computed as:

$$u' = u'_{\Delta_e}\left(\frac{l_t}{\Delta_e}\right)^{1/3} \tag{7.57}$$

where u' is the RMS velocity at the integral length scale l_t.

$$\begin{aligned}\Gamma_0 &= 0.75\exp\left\{-1.2/\left(u'_{\Delta_e}/s_l^0\right)^{0.3}\right\}\left(\Delta_e/\delta_l^0\right)^{2/3}\\ \Gamma_1 &= 0.75\exp\left\{-1.2/\left(u'_{\Delta_e}/s_l^0\right)^{0.3}\right\}\left(\Delta_e/\delta_l^1\right)^{2/3}\end{aligned} \tag{7.58}$$

Here, the superscript 0 refers to the flame without thickening, and the superscript 1 refers to the thickened flame.

When the flow is laminar, u'_{Δ_e} equals 0, and the Colin efficiency function is reduced to 1. When the turbulent velocity fluctuations vary, such that

$$\alpha\Gamma_0\frac{u'_{\Delta_e}}{s_l^0} >> 1 \quad and \quad \alpha\Gamma_1\frac{u'_{\Delta_e}}{s_l^0} >> 1 \tag{7.59}$$

the Colin efficiency function approaches a theoretical maximum limit:

$$E_{MAX} = \frac{\Gamma_0}{\Gamma_1} \tag{7.60}$$

The Charlette efficiency function [100] (p. 1058), [684] (p. 1092) is computed as follows:

$$E = \left(1 + min\left[\left(\frac{\Delta_e}{\delta_l} - 1\right), \Gamma_{\Delta_e}\left(\frac{\Delta_e}{\delta_l}, \frac{u'_{\Delta_e}}{s_l^0}, Re_{\Delta_e}\right)\frac{u'_{\Delta_e}}{s_l^0}\right]\right)^{\beta_{cha}} \tag{7.61}$$

where

Δ_e = filter width that corresponds to the thickened flame, which is usually an order of magnitude higher than the physical grid length scale. The default value is 10 times the local cell grid spacing.

δ_l = laminar flame thickness.

u'_{Δ_e} = sub grid scale turbulent velocity.

s_l^0 = laminar flame speed.

Re_{Δ_e} = sub grid Reynolds number at the filter scale.

β_{cha} = exponent set to a constant value of 0.5.

Γ_{Δ_e} is calculated as:

$$\Gamma_{\Delta_e} = \left[\left\{\left(f_u^{-a} + f_\Delta^{-a}\right)^{-1/a}\right\}^{-b} + f_{Re}^{-b}\right]^{-1/b} \tag{7.62}$$

with b = 1.4, and

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$a=0.6+0.2\exp\left(-0.1\frac{u'_{\Delta_e}}{s_l^0}\right)-0.2\exp\left(-0.01\frac{\Delta_e}{\delta_l}\right) \tag{7.63}$$

$$f_u=4\left(\frac{27C_k}{110}\right)^{1/2}\left(\frac{18C_k}{55}\right)\left(\frac{u'_{\Delta_e}}{s_l^0}\right)^2 \tag{7.64}$$

$$f_\Delta=\left[\frac{27\,C_k\,\pi^{4/3}}{110}\left\{\left(\frac{\Delta_e}{\delta_l}\right)^{4/3}-1\right\}\right]^{1/2} \tag{7.65}$$

$$f_{Re}=\left[\frac{9}{55}\exp\left(-\frac{3}{2}C_k\,\pi^{4/3}\,Re_{\Delta_e}^{-1}\right)\right]^{1/2}Re_{\Delta_e}^{1/2} \tag{7.66}$$

In the above equations, C_k is the Kolmogorov constant with a value of 1.5.

All species reaction rates are multiplied by E/F. The effective species diffusivities (and thermal conductivity) in each cell are dynamically determined as:

$$D_{eff}=D_{lam}\,E(1+(F-1)\Omega)+D_{turb}(1-\Omega) \tag{7.67}$$

where D_{lam} is the molecular (laminar) diffusivity and D_{turb} is the turbulent diffusivity. D_{lam} may be computed with any of the available methods in Ansys Fluent, including kinetic theory and User Defined Functions (UDFs). Since the kinetic coefficients of 1-step reactions are invariably adjusted to capture the actual laminar flame speed, the same transport properties that are used in this tuning simulation should be used in the TFM simulation. In the narrow band around the flame where Ω is one, the turbulent diffusivities are switched off and the molecular diffusivities are enhanced by a factor EF. Away from the flame where Ω is zero, the effective diffusivity is the non-thickened value of $D_{lam}+D_{turb}$.

7.1.1.2.7. The Relaxation to Chemical Equilibrium Model

When using the finite-rate (FR) chemistry, or when using eddy-dissipation (ED), finite-rate/eddy-dissipation (FR/ED) or EDC model, as described above, chemical species evolve according to the prescribed kinetic mechanism. In the Relaxation to Chemical Equilibrium model, the species composition is driven to its equilibrium state. The reaction source term in the i^{th} mean species conservation equation (Equation 7.1 (p. 238)), is modeled as

$$R_i=\rho\frac{Y_i^{eq}-Y_i}{\tau_{char}} \tag{7.68}$$

where,

ρ = the density

Y_i = the mean mass fraction of species i

superscript eq denotes chemical equilibrium

τ_{char} = a characteristic time-scale

Equation 7.68 (p. 254) implies that species react toward their chemical equilibrium state over the characteristic time, τ_{char}, as in the Characteristic Time model [318] (p. 1071). Since chemical equilibrium does not depend on reactions or reaction rates, for a given τ_{char}, the reaction source term in Equation 7.68 (p. 254) is independent of the reaction mechanism.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

The Relaxation to Chemical Equilibrium model is an option available for any of the turbulence-chemistry interaction options. When no turbulence-chemistry model is used, the characteristic time is calculated as

$$\tau_{char}=\tau_{flow}+\tau_{chem} \tag{7.69}$$

where τ_{flow} is a convection/diffusion time-scale in a cell. τ_{chem} is a cell chemical time-scale modeled as

$$\tau_{chem}=c_{chem}\min_f\left(\frac{Y_f}{dY_f/dt}\right) \tag{7.70}$$

In Equation 7.70 (p. 255), c_{chem} is a model constant (default of 1), index f denotes the user-specified fuel species, Y_f is the fuel species mass fraction, and dY_f/dt is the rate of change of the fuel species mass fraction, while $\min$ denotes the minimum over all specified fuel species. Hence, τ_{chem} represents a chemical ignition time-scale for the fuel species, and is used to prevent a premixed flame from auto-igniting, similar to the FR/ED model. Note that a kinetic mechanism is required to evaluate τ_{chem}.

For the ED model, the characteristic time-scale is evaluated as $\tau_{char}=\tau_{turb}$, where the turbulent time-scale is $\tau_{turb}=k/(\epsilon A)$, and the default turbulent rate constant is A=4. For the FR/ED model, the characteristic time-scale is calculated as $\tau_{char}=\tau_{turb}+\tau_{chem}$. Typically, the ED and FR/ED models employ a 1-step reaction, or a 2-step reaction where hydrocarbons pyrolize to CO, which then oxidizes to CO_2. The Relaxation to Chemical Equilibrium model can be considered as an extension of the ED type models where species react towards chemical equilibrium instead of complete reaction. The model should provide more accurate predictions of intermediate species such as CO and radicals required for NOx modeling such as O and OH.

Since chemical equilibrium calculations are typically less computationally expensive than detailed chemistry simulations, the Relaxation to Chemical equilibrium can be used to provide a good initial condition for full-kinetic steady-state simulations. The model can also be used in lieu of the Eddy-Dissipation model where the solution tends to chemical equilibrium instead of complete reaction.

The assumption of chemical equilibrium can lead to large errors in rich zones for hydrocarbon fuels. Ansys Fluent offers the option of reducing the reaction rate in Equation 7.68 (p. 254) by increasing the characteristic time-scale τ_{char} in rich zones. With this option, the local equivalence ratio in a cell is calculated as

$$\phi=\frac{2X_C+X_H/2}{X_O} \tag{7.71}$$

where X_O, X_H, and X_C denote the atomic molar fraction of oxygen, hydrogen and carbon. The time-scale, τ_{char}, is then multiplied by a factor $e^{\beta(\phi-\phi_{rich})}$ when the local equivalence ratio ϕ exceeds the specified rich equivalence ratio ϕ_{rich}. The default rich equivalence ratio is 1, and the default exponential factor, β, is 2. The option to enable the slow reaction rate in rich mixtures is available in the text interface.

Since chemical equilibrium calculations can consume computational resources, ISAT tabulation is the recommended (and default) solution method. The initial Ansys Fluent iterations speed up significantly as the table is built up with sufficient entries and the majority of queries to the table

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

are interpolated. Since chemical equilibrium compositions are uniquely determined by the initial temperature, pressure and elemental compositions of the mixture, the ISAT table dimensions are $N_{elements}+1$ for isobaric systems. In most combustion cases, the number of elements is much smaller than the number of chemical species, and the ISAT table is built up relatively quickly. The **Direct Integration** option may also be selected, in which case equilibrium calculations will always be performed at all cells.

7.1.1.3. Finite-Rate Chemistry with the Two-Temperature Model

At hypersonic speeds, chemical reactions such as dissociation and recombination can have large impact on the flow. If the flow characteristic time and the chemical reaction time are of approximately the same order, the fluid element does not reside at one location long enough to bring the local chemical reactions to equilibrium, and therefore, the flow could be in locally chemical non-equilibrium. The two-temperature model can be used with the finite-rate chemistry model to simulate these thermal-chemical non-equilibrium flows.

When the two-temperature model is coupled with the finite-rate chemistry model to describe the thermal-chemical non-equilibrium phenomena in hypersonic flows, Equation 7.23 (p. 245) takes the form:

$$k_{f,r}=A_r T_c^{\beta_r} e^{-E_r/RT_c} \tag{7.72}$$

where $T_c=T_{tr}^{\alpha}T_{ve}^{1-\alpha}$ is the control temperature. The coefficient $\alpha=0.7$ for dissociation reactions, to account for the fact that dissociative reactions are greatly influenced by the vibrational modes. For exchange reactions, the coefficient $\alpha=1$.

For details on specifying the control temperature coefficient, see Using the Two-Temperature Model in the *Fluent User's Guide*.

7.1.2. Wall Surface Reactions and Chemical Vapor Deposition

For gas-phase reactions, the reaction rate is defined on a volumetric basis and the net rate of creation and destruction of chemical species becomes a source term in the species conservation equations. For surface reactions, the rate of adsorption and desorption is governed by both chemical kinetics and diffusion to and from the surface. Wall surface reactions therefore create sources and sinks of chemical species in the gas phase, as well as on the reacting surface.

Theoretical information about wall surface reactions and chemical vapor deposition is presented in this section. Information can be found in the following sections:

- 7.1.2.1. Surface Coverage Reaction Rate Modification
- 7.1.2.2. Reaction-Diffusion Balance for Surface Chemistry
- 7.1.2.3. Slip Boundary Formulation for Low-Pressure Gas Systems

For more information about using wall surface reactions and chemical vapor deposition, see Wall Surface Reactions and Chemical Vapor Deposition in the User's Guide.

Consider the r^{th} wall surface reaction written in general form as follows:

$$\sum_{i=1}^{N_g} g_{i,r}^{'} G_i + \sum_{i=1}^{N_b} b_{i,r}^{'} B_i + \sum_{i=1}^{N_s} s_{i,r}^{'} S_i \quad \Leftrightarrow \frac{k_{f,r}}{k_{b,r}} \Leftrightarrow \quad \sum_{i=1}^{N_g} g_{i,r}^{''} G_i + \sum_{i=1}^{N_b} b_{i,r}^{''} B_i + \sum_{i=1}^{N_s} s_{i,r}^{''} S_i \tag{7.73}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

where G_i, B_i, and S_i represent the gas phase species, the bulk (or solid) species, and the surface-adsorbed (or site) species, respectively. N_g, N_b, and N_s are the total numbers of these species. $g'_{i,r}$, $b'_{i,r}$, and $s'_{i,r}$ are the stoichiometric coefficients for each reactant species i, and $g''_{i,r}$, $b''_{i,r}$, and $s''_{i,r}$ are the stoichiometric coefficients for each product species i. $k_{f,r}$ and $k_{b,r}$ are the forward and backward rate of reactions, respectively.

The summations in Equation 7.73 (p. 256) are for all chemical species in the system, but only species involved as reactants or products will have nonzero stoichiometric coefficients. Hence, species that are not involved will drop out of the equation.

The rate of the r^{th} reaction is

$$\mathfrak{R}_r = k_{f,r}\left(\prod_{i=1}^{N_g}[C_i]_{wall}^{\eta'_{i,g,r}}\right)\left(\prod_{j=1}^{N_s}[S_j]_{wall}^{\eta'_{j,s,r}}\right) - k_{b,r}\left(\prod_{i=1}^{N_g}[C_i]_{wall}^{\eta''_{i,g,r}}\right)\left(\prod_{j=1}^{N_s}[S_j]_{wall}^{\eta''_{j,s,r}}\right) \tag{7.74}$$

where $[\,]_{wall}$ represents molar concentrations of surface-adsorbed species on the wall. $\eta'_{i,g,r}$ and $\eta''_{i,g,r}$ are the rate exponents for the i^{th} gaseous species as reactant and product, respectively, in the reaction. The variables $\eta'_{j,s,r}$ and $\eta''_{j,s,r}$ are the rate exponents for the j^{th} site species as reactant and product in the reaction. It is assumed that the reaction rate does not depend on concentrations of the bulk (solid) species. From this, the net molar rate of production or consumption of each species i is given by

$$\hat{R}_{i,gas} = \sum_{r=1}^{N_{rxn}}\left(g''_{i,r} - g'_{i,r}\right)\mathfrak{R}_r \quad i=1,2,3,\ldots,N_g \tag{7.75}$$

$$\hat{R}_{i,bulk} = \sum_{r=1}^{N_{rxn}}\left(b''_{i,r} - b'_{i,r}\right)\mathfrak{R}_r \quad i=1,2,3,\ldots,N_b \tag{7.76}$$

$$\hat{R}_{i,site} = \sum_{r=1}^{N_{rxn}}\left(s''_{i,r} - s'_{i,r}\right)\mathfrak{R}_r \quad i=1,2,3,\ldots,N_s \tag{7.77}$$

The forward rate constant for reaction r ($k_{f,r}$) is computed using the Arrhenius expression,

$$k_{f,r} = A_r T^{\beta_r} e^{-E_r/RT} \tag{7.78}$$

where,

A_r = pre-exponential factor (consistent units)

β_r = temperature exponent (dimensionless)

E_r = activation energy for the reaction (J/kmol)

R = universal gas constant (J/kmol-K)

You (or the database) will provide values for $g'_{i,r}$, $g''_{i,r}$, $b'_{i,r}$, $b''_{i,r}$, $s'_{i,r}$, $s''_{i,r}$, β_r, A_r, and E_r.

To include the mass transfer effects and model heat release, refer to Including Mass Transfer To Surfaces in the Continuity Equation, Wall Surface Mass Transfer Effects in the Energy Equation, and Modeling the Heat Release Due to Wall Surface Reactions in the User's Guide.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

If the reaction is reversible, the backward rate constant for reaction r, $k_{b,r}$ is computed from the forward rate constant using the following relation:

$$k_{b,r}=\frac{k_{f,r}}{K_r} \tag{7.79}$$

where K_r is the equilibrium constant for r^{th} reaction computed from

$$K_r=\exp\left(\frac{\Delta S_r}{R}-\frac{\Delta H_r}{RT}\right)\left(\frac{p_{atm}}{RT}\right)^{\sum_{i=1}^{N_g}\left(\upsilon''_{i,r}-\upsilon'_{i,r}\right)}\prod_{k=1}^{N_{types}}\left(\rho_s\right)_k^{\sum_{j=1}^{N_{sk}}\left(\upsilon''_{j,k,r}-\upsilon'_{j,k,r}\right)} \tag{7.80}$$

where p_{atm} denotes the atmospheric pressure (101325 Pa). The term within the exponential function represents the change in the Gibbs free energy, and its components are computed per Equation 7.26 (p. 245) and Equation 7.27 (p. 245).

N_{types} is the number of different types of sites, $\left(\rho_s\right)_k$ is the site density of site type k. $\upsilon'_{j,k,r}$ and $\upsilon''_{j,k,r}$ are the stoichiometric coefficients of the j^{th} site species of type k in reaction r.

7.1.2.1. Surface Coverage Reaction Rate Modification

Ansys Fluent has the option to modify the surface reaction rate as a function of species site coverages. In such cases, the forward rate constant for the r^{th} reaction is evaluated as,

$$k_{f,r}=A_r T^{\beta_r} e^{-E_r/RT}\prod_{k_{site}}\left(10^{Z_k\eta_{k,r}}\right)\left(Z_k^{\mu_{k,r}}\right)\left(e^{-\varepsilon_{k,r}Z_k/RT}\right) \tag{7.81}$$

In Equation 7.81 (p. 258), the three surface coverage rate modification parameters for species k in reaction r are $\eta_{k,r}$, $\mu_{k,r}$ and $\varepsilon_{k,r}$. These parameters default to zero for reaction species that are not surface rate modifying. The surface (coverage) site fraction, Z_k is the fraction of surface sites covered by species k, and is defined as,

$$Z_k=\left[S_k\right]/\rho_{site} \tag{7.82}$$

where $\left[S_k\right]$ is the surface site concentration and ρ_{site} is the surface site density (see Equation 7.87 (p. 259)).

7.1.2.2. Reaction-Diffusion Balance for Surface Chemistry

Reactions at surfaces change gas-phase, surface-adsorbed (site) and bulk (solid) species. On reacting surfaces, the mass flux of each gas species due to diffusion and convection to/from the surface is balanced with its rate of consumption/production on the surface,

$$\rho_{wall}D_{i,wall}\frac{\partial Y_{i,wall}}{\partial n}-\dot{m}_{dep}Y_{i,wall}=M_{w,i}\hat{R}_{i,gas}\qquad i=1,2,3,\ldots,N_g \tag{7.83}$$

$$\frac{\partial\left[S_i\right]_{wall}}{\partial t}=\hat{R}_{i,site}\quad i=1,2,3,\ldots,N_s \tag{7.84}$$

The mass fraction at the wall $Y_{i,wall}$ is related to concentration by

$$\left[G_i\right]_{wall}=\frac{\rho_{wall}Y_{i,wall}}{M_{w,i}} \tag{7.85}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$\dot{m}_{dep}$ is the net rate of mass deposition or etching as a result of surface reaction; that is,

$$\dot{m}_{dep} = \sum_{i=1}^{N_b} M_{w,i} \hat{R}_{i,bulk} \tag{7.86}$$

$[S_i]_{wall}$ is the site species concentration at the wall, and is defined as

$$[S_i]_{wall} = \rho_{site} Z_i \tag{7.87}$$

where ρ_{site} is the site density and Z_i is the site coverage of species i.

Equation 7.83 (p. 258) and Equation 7.84 (p. 258) are solved for the dependent variables $Y_{i,wall}$ and Z_i using a point-by-point coupled Newton solver. When the Newton solver fails, Equation 7.83 (p. 258) and Equation 7.84 (p. 258) are solved by time marching in an ODE solver until convergence. If the ODE solver fails, reaction-diffusion balance is disabled, $Y_{i,wall}$ is assumed equal to the cell-center value $Y_{i,cell}$, and only the site coverages Z_i are advanced in the ODE solver to convergence. You can manually disable reaction-diffusion balance with the text interface command `define/models/species/reaction-diffusion-balance`.

The effective gas-phase reaction source terms are then available for solution of the gas-phase species transport Equation 7.1 (p. 238).

The diffusion term in Equation 7.83 (p. 258) is calculated as the difference in the species mass fraction at the cell center and the wall-face center, divided by the normal distance between these center points. Ansys Fluent models the reactions that take place inside catalyst structures by introducing the **Surface Area Washcoat Factor**. Generally, a washcoat is a carrier in which the catalytic materials are suspended. These materials increase the surface area compared to the bare substrate by forming a rough and irregular surface; increasing the catalytically active reacting area available for surface reaction. To utilize the available reacting surface, Ansys Fluent provides an option to define a washcoat factor. This coefficient is multiplied by the reacting surface to calculate the available reacting surface which modifies the surface reaction rates linearly. Ansys Fluent can also model surface chemistry on unresolved walls using a porous media model. This model is appropriate for catalytic tube-banks or porous foam matrices where it is not feasible to resolve the individual walls. When surface reaction is enabled in porous cell zones, the **Surface-to-Volume Ratio** must be specified. The wall normal distance required for the diffusion term in Equation 7.83 (p. 258) is calculated as the inverse of the surface-to-volume ratio.

7.1.2.3. Slip Boundary Formulation for Low-Pressure Gas Systems

Most semiconductor fabrication devices operate far below atmospheric pressure, typically only a few millitorrs. At such low pressures, the fluid flow is in the slip regime and the normally used no-slip boundary conditions for velocity and temperature are no longer valid.

The Knudsen number, denoted Kn, and defined as the ratio of mean free path to a characteristic length scale of the system, is used to quantify continuum flow regimes. Since the mean free path increases as the pressure is lowered, the high end of Kn values represents *free molecular flow* and the low end the continuum regime. The range in between these two extremes is called the slip regime ($0.01<Kn<0.1$) [63] (p. 1056) In the slip regime, the gas-phase velocity at a solid surface differs from the velocity at which the wall moves, and the gas temperature at the surface differs from the wall temperature. Maxwell's models are adopted for these physical phenomena in Ansys Fluent for their simplicity and effectiveness.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

- velocity slip

$$U_w - U_g = \left(\frac{2-\alpha_v}{\alpha_v}\right) KnL_c \frac{\partial U}{\partial n} \approx \left(\frac{2-\alpha_v}{\alpha_v}\right) \frac{\lambda}{\delta} (U_g - U_c) \tag{7.88}$$

$$V_g \equiv \left(\vec{V} \cdot \vec{n}\right)_g = V_w \tag{7.89}$$

Here, U and V represent the velocity components that are tangential and normal to the wall, respectively. The subscripts g, w and c indicate gas, wall and cell-center velocities. δ is the distance from cell center to the wall. L_c is the characteristic length. α_v is the momentum accommodation coefficient of the gas mixture and its value is calculated as mass-fraction weighted average of each gas species in the system.

$$\alpha_v = \sum_{i=1}^{N_g} Y_i \alpha_i \tag{7.90}$$

The mean free path, λ, is computed as follows:

$$\lambda = \frac{k_B T}{\sqrt{2}\pi\sigma^2 p} \tag{7.91}$$

$$\sigma = \sum_{i=1}^{N_g} Y_i \sigma_i \tag{7.92}$$

σ_i is the Lennard-Jones characteristic length of species i. k_B is the Boltzmann constant, $1.38066 \times 10^{-23} J/K$.

Equation 7.88 (p. 260) and Equation 7.89 (p. 260) indicate that while the gas velocity component normal to the wall is the same as the wall normal velocity, the tangential components slip. The values lie somewhere between the cell-center and the wall values. These two equations can be combined to give a generalized formulation:

$$\vec{V}_g = \frac{\vec{V}_w + \frac{k}{\delta}\left[\left(\vec{V}_w \cdot \vec{n}\right)\vec{n} + \vec{V}_c - \left(\vec{V}_c \cdot \vec{n}\right)\vec{n}\right]}{1 + \frac{k}{\delta}} \tag{7.93}$$

where

$$k \equiv \lambda\left(\frac{2-\alpha_v}{\alpha_v}\right) \tag{7.94}$$

- temperature jump

$$T_w - T_g = 2\left(\frac{2-\alpha_T}{\alpha_T}\right) KnL_c \frac{\partial T}{\partial n} \approx 2\left(\frac{2-\alpha_T}{\alpha_T}\right) \frac{\lambda}{\delta} (T_g - T_c) \tag{7.95}$$

or equivalently

$$T_g = \frac{T_w + \beta T_c}{1 + \beta} \tag{7.96}$$

where

$$\beta = \frac{2(2-\alpha_T)\lambda}{\alpha_T \delta} \tag{7.97}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

α_T is the thermal accommodation coefficient of the gas mixture and is calculated as $\alpha_T = \sum Y_i \alpha_{T,i}$.

Important:

The low-pressure slip boundary formulation is available only with the pressure-based solver.

7.1.3. Particle Reactions

Theoretical information about particle reactions is presented in the following sections:

7.1.3.1. Combusting Particle Surface Reactions

7.1.3.2. Multicomponent Particles with Chemical Reactions

7.1.3.1. Combusting Particle Surface Reactions

As described in The Multiple Surface Reactions Model (p. 484), it is possible to define multiple particle surface reactions to model the surface combustion of a combusting discrete-phase particle. This section provides theoretical background about particle surface reactions. Information can be found in the following sections:

7.1.3.1.1. General Description

7.1.3.1.2. Ansys Fluent Model Formulation

7.1.3.1.3. Extension for Stoichiometries with Multiple Gas Phase Reactants

7.1.3.1.4. Solid-Solid Reactions

7.1.3.1.5. Solid Decomposition Reactions

7.1.3.1.6. Solid Deposition Reactions

7.1.3.1.7. Gaseous Solid Catalyzed Reactions on the Particle Surface

For more information about using particle surface reactions, see Combusting Particle Surface Reactions in the User's Guide.

7.1.3.1.1. General Description

The relationships for calculating char particle burning rates are presented and discussed in detail by Smith [606] (p. 1088). The particle reaction rate, $\mathfrak{R}$ ($\mathrm{kg/m^2\text{-}s}$), can be expressed as

$$\mathfrak{R} = D_0 \left(C_g - C_s \right) = R_c \left(C_s \right)^N \tag{7.98}$$

where,

D_0 = bulk diffusion coefficient (m/s)

C_g = mean reacting gas species concentration in the bulk ($\mathrm{kg/m^3}$)

C_s = mean reacting gas species concentration at the particle surface ($\mathrm{kg/m^3}$)

R_c = chemical reaction rate coefficient (units vary)

N = apparent reaction order (dimensionless)

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

In Equation 7.98 (p. 261), the concentration at the particle surface, C_s, is not known, so it should be eliminated, and the expression is recast as follows:

$$\mathcal{R}=R_c\left[C_g-\frac{\mathcal{R}}{D_0}\right]^N \tag{7.99}$$

This equation has to be solved by an iterative procedure, with the exception of the cases when $N=1$ or $N=0$. When $N=1$, Equation 7.99 (p. 262) can be written as

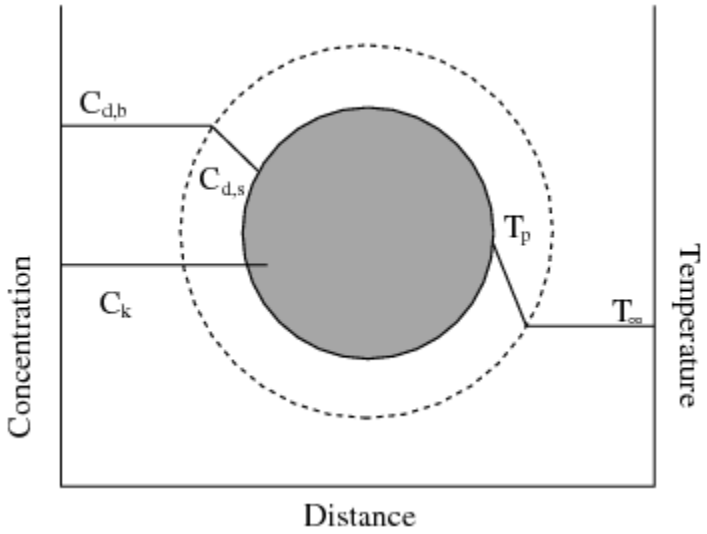
$$\mathcal{R}=\frac{C_g R_c D_0}{D_0+R_c} \tag{7.100}$$

In the case of $N=0$, if there is a finite concentration of reactant at the particle surface, the solid depletion rate is equal to the chemical reaction rate. If there is no reactant at the surface, the solid depletion rate changes abruptly to the diffusion-controlled rate. In this case, however, Ansys Fluent will always use the chemical reaction rate for stability reasons.

7.1.3.1.2. Ansys Fluent Model Formulation

A particle undergoing an exothermic reaction in the gas phase is shown schematically in Figure 7.1: A Reacting Particle in the Multiple Surface Reactions Model (p. 262). T_p and T_∞ are the temperatures in Equation 12.159 (p. 485).

Figure 7.1: A Reacting Particle in the Multiple Surface Reactions Model



Based on the analysis above, Ansys Fluent uses the following equation to describe the rate of reaction r of a particle surface species j with the gas phase species n. The reaction stoichiometry of reaction r in this case is described by

$$\text{particle species}\, j(\text{s})+\text{gas phase species}\, n\rightarrow\text{products} \tag{7.101}$$

and the rate of reaction is given as

$$\bar{\mathcal{R}}_{j,r}=A_p\eta_r Y_j\mathcal{R}_{j,r} \tag{7.102}$$

$$\mathcal{R}_{j,r}=\mathcal{R}_{kin,r}p_n^{N_r}\left(1-\frac{\mathcal{R}_{j,r}}{p_n D_{0,r}}\right)^{N_r} \tag{7.103}$$

where,

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$\bar{\mathfrak{R}}_{j,r}$ = rate of particle surface species depletion (kg/s)

A_p = particle surface area (m^2)

Y_j = mass fraction of surface species j in the particle

η_r = effectiveness factor (dimensionless)

$\mathfrak{R}_{j,r}$ = rate of particle surface species reaction per unit area ($kg/m^2/s$)

p_n = bulk partial pressure of the gas phase species (Pa)

$D_{0,r}$ = diffusion rate coefficient for reaction r ($kg/m^2/s/Pa$)

$\mathfrak{R}_{kin,r}$ = kinetic rate of reaction r (units vary)

N_r = apparent order of reaction r

The effectiveness factor, η_r, is related to the surface area, and can be used in each reaction in the case of multiple reactions. $D_{0,r}$ is given by

$$D_{0,r}=C_{1,r}\frac{\left[\left(T_p+T_\infty\right)/2\right]^{0.75}}{d_p} \tag{7.104}$$

The kinetic rate of reaction r is defined as

$$\mathfrak{R}_{kin,r}=A_r T_p^{\beta_r} e^{-\left(E_r/RT_p\right)} \tag{7.105}$$

The rate of the particle surface species depletion for reaction order $N_r=1$ is given by

$$\bar{\mathfrak{R}}_{j,r}=A_p\eta_r Y_j p_n\frac{\mathfrak{R}_{kin,r}D_{0,r}}{D_{0,r}+\mathfrak{R}_{kin,r}} \tag{7.106}$$

For reaction order $N_r=0$,

$$\bar{\mathfrak{R}}_{j,r}=A_p\eta_r Y_j\mathfrak{R}_{kin,r} \tag{7.107}$$

7.1.3.1.3. Extension for Stoichiometries with Multiple Gas Phase Reactants

When more than one gas phase reactant takes part in the reaction, the reaction stoichiometry must be extended to account for this case:

$$\begin{aligned}\text{particle species}j(\text{s}) \quad &+\text{gas phase species1}+\text{gas phase species2}+\ldots\\ &+\text{gas phase species}n_{max}\rightarrow\text{products}\end{aligned} \tag{7.108}$$

To describe the rate of reaction r of a particle surface species j in the presence of n_{max} gas phase species n, it is necessary to define the diffusion-limited species for each solid particle reaction, that is, the species for which the concentration gradient between the bulk and the particle surface is the largest. For the rest of the species, the surface and the bulk concentrations are assumed to be equal. The concentration of the diffusion-limited species is shown as $C_{d,b}$ and $C_{d,s}$ in Figure 7.1: A Reacting Particle in the Multiple Surface Reactions Model (p. 262), and the concentrations of all other species are denoted as C_k. For stoichiometries with multiple gas phase reactants, the bulk partial pressure p_n in Equation 7.103 (p. 262) and Equation 7.106 (p. 263) is the bulk partial pressure of the diffusion-limited species, $p_{r,d}$ for reaction r.

The kinetic rate of reaction r is then defined as

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$\mathcal{R}_{kin,r} = \frac{A_r T^{\beta_r} e^{-(E_r/RT)}}{\left(p_{r,d}\right)^{N_{r,d}}} \prod_{n=1}^{n_{max}} p_n^{N_{r,n}} \tag{7.109}$$

where,

p_n = bulk partial pressure of gas species n

$N_{r,n}$ = reaction order in species n

When this model is enabled, the constant $C_{1,r}$ (Equation 7.104 (p. 263)) and the effectiveness factor η_r (Equation 7.102 (p. 262)) are entered in the **Reactions** dialog box (see User Inputs for Particle Surface Reactions in the User's Guide).

7.1.3.1.4. Solid-Solid Reactions

Reactions involving only particle surface reactants can be modeled, provided that the particle surface reactants and products exist on the same particle.

$$\text{particle species1(s)} + \text{particle species2(s)} + ... \rightarrow \text{products}$$

The reaction rate for this case is given by Equation 7.107 (p. 263).

7.1.3.1.5. Solid Decomposition Reactions

The decomposition reactions of particle surface species can be modeled.

$$\begin{aligned} \text{particle species1(s)} + \text{particle species2(s)} + ... \quad & + \text{particle species} n_{max}\text{(s)} \\ & \rightarrow \text{gas species} j + \text{products} \end{aligned} \tag{7.110}$$

The reaction rate for this case is given by Equation 7.102 (p. 262) – Equation 7.109 (p. 264), where the diffusion-limited species is now the gaseous product of the reaction. If there are more than one gaseous product species in the reaction, it is necessary to define the diffusion-limited species for the particle reaction as the species for which the concentration gradient between the bulk and the particle surface is the largest.

7.1.3.1.6. Solid Deposition Reactions

The deposition reaction of a solid species on a particle can be modeled with the following assumptions:

$$\text{gas species1} + \text{gas species2} + ... + \text{gas species} n_{max} \rightarrow \text{solid species} j\text{(s)} + \text{products} \tag{7.111}$$

The theoretical analysis and Equation 7.102 (p. 262) – Equation 7.109 (p. 264) are applied for the surface reaction rate calculation, with the mass fraction of the surface species set to unity in Equation 7.102 (p. 262), Equation 7.106 (p. 263), and Equation 7.107 (p. 263).

In Ansys Fluent, for the particle surface species to be deposited on a particle, a finite mass of the species must already exist in the particle. This allows for activation of the deposition reaction selectively to particular injection particles. It follows that, to initiate the solid species deposition reaction on a particle, the particle must be defined in the Set Injection Properties Dialog Box (or Set Multiple Injection Properties Dialog Box) to contain a small mass fraction of the solid species

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

to be deposited. For details on defining the particle surface species mass fractions, see Using the Multiple Surface Reactions Model for Discrete-Phase Particle Combustion in the User's Guide.

7.1.3.1.7. Gaseous Solid Catalyzed Reactions on the Particle Surface

Reactions of gaseous species catalyzed on the particle surface can also be modeled following Equation 7.102 (p. 262) – Equation 7.109 (p. 264) for the surface reaction rate calculation, with the mass fraction of the surface species set to unity in Equation 7.102 (p. 262), Equation 7.106 (p. 263), and Equation 7.107 (p. 263). To apply this type of reaction, see Modeling Gaseous Solid Catalyzed Reactions in the User's Guide. For details on defining the particle surface species mass fractions, see Using the Multiple Surface Reactions Model for Discrete-Phase Particle Combustion in the User's Guide.

7.1.3.2. Multicomponent Particles with Chemical Reactions

Components of multicomponent particles can participate in chemical reactions. Reactions can take place between particle components, or with a gas phase component, and particle components are allowed to evaporate and react in sequence or simultaneously. Several reaction models are available as built-in functionality as well as a user defined option. The multicomponent particle may contain reacting, volatile, or inert components that do not participate in any reactions.

Similarly to the combusting particle solid components, the multicomponent particle components can participate in chemical reactions as described below.

- Particle species reacts with one or more gas-phase species
- Particle species reacts with one or more particle species
- Particle species decomposes to gas phase species
- A gas phase species is converted to a particle species and is deposited on the particle
- Catalytic reactions where the gas phase species react on the particle surface without consuming a particle species

For these reaction types, the following reaction models are used in Ansys Fluent:

- kinetics/diffusion

  The reaction rate is calculated according to Ansys Fluent Model Formulation (p. 262).

- surface-kinetics

  The reaction rate is calculated as:

$$R_{i,j} = A_p \eta k Y_i \prod_{n=1}^{n_{max}} p_n{}^{k_n} \tag{7.112}$$

  where:

  $R_{i,j}$ = rate of reaction j of particle component i (kg/s)

  A_p = surface area of the particle (m^2)

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

η = effectiveness factor

k = reaction rate coefficient

Y_i = mass fraction of particle component i

p_n = partial pressure of gas reactant n

k_n = reaction order for gas reactant n

The reaction rate coefficient k is expressed as:

$$k=AT^{\beta}e^{(-E/RT)} \tag{7.113}$$

where:

A = pre-exponential factor

T = = temperature of the particle (K)

β = exponent on temperature

E = activation energy (J/Kg-mol/K)

R = ideal gas constant

- volume-kinetics

  The reaction rate is calculated as:

$$R_{i,j} = V_p M_i k \prod_{n=1}^{n_{max}} C_n^{k_n} \prod_{i=1}^{i_{max}} C_i^{k_i} \tag{7.114}$$

  where:

  $R_{i,j}$ = rate of reaction j of particle component i (kg/s)

  V_p = particle volume (m^3)

  M_i = molecular weight of particle component i (Kg/Kg-mol)

  k = reaction rate coefficient given by Equation 7.113 (p. 266)

  C_n and C_i = concentrations of gas reactant n and particle reactant i, respectively (Kg-mol/m^3)

  k_n and k_i = reaction orders for gas reactant n and particle reactant i, respectively

- phase-change

  In this reaction model, particle component can evaporate con-currently with the other reacting components. Only one solid/condensed particle material is allowed in the reaction stoichiometry as reactant and one gas material as product. The phase-change rate of particle component i to gas component n is calculated according to:

$$R_{i,v} = A_p M_i k_v \left(C_i - C_n\right) \tag{7.115}$$

  $R_{i,v}$ = rate of phase-change of particle component i (kg/s)

  A_p = surface area of the particle (m^2)

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

M_i = molecular weight of particle component i (Kg/Kg-mol)

k_v = mass transfer coefficient (m/s)

C_i and C_n = concentrations of particle reactant i at the particle surface and gas phase species n, respectively (Kg-mol/m^3)

In addition to the above reaction models, you can define your own reaction rate using a user-defined function. For more information on using these models, see Multicomponent Particles with Chemical Reactions in the *Fluent User's Guide*.

7.1.4. Electrochemical Reactions

This section presents the theoretical background for modeling electrochemical reactions in Ansys Fluent. The following topics are discussed:

7.1.4.1. Overview

7.1.4.2. Electrochemical Reaction Model Theory

7.1.4.1. Overview

An electrochemical charge-transfer reaction is a chemical reaction between neutral species, charged ions, and electrons. Phenomena associated with electrochemistry include batteries, fuel cells, corrosion and electro-deposition. Charge-transfer reactions occur at phase-interfaces between electrodes (anode or cathode) and an electrolyte. This interface is referred to as a Faradaic surface.

7.1.4.2. Electrochemical Reaction Model Theory

The general form of the r^{th} electrochemical reaction is:

$$\sum_{k=1}^{N} \nu'_{k,r} M_k^{z_k} \leftrightarrow \sum_{k=1}^{N} \nu''_{k,r} M_k^{z_k} + \left(\sum_{k=1}^{N} \left(\nu''_{k,r} - \nu'_{k,r} \right) z_k \right) e^- \tag{7.116}$$

where:

M_k represents liquid or solid species k

N is the total number of species

$\nu'_{k,r}$ and $\nu''_{k,r}$ are the stoichiometric coefficients of the k^{th} species as reactant and product in the reaction r, respectively

z_k is the charge number of the k^{th} species

e^- represents electron

For neutral species, the charge number z_k is zero.

In Ansys Fluent, the charge-transfer reaction rate for the electrochemical reaction r at Faradaic interfaces is determined by the Butler-Volmer equation:

$$i_{F,r} = i_{o,r} \, exp\left[-\frac{E}{R} \left(\frac{1}{T} - \frac{1}{T_{ref}} \right) \right] \prod_{k=1}^{N} \left(\frac{C_{k,r}}{C_{k,r}^{ref}} \right)^{\nu_{K,r}} \left[\exp\left(\frac{\alpha_{a,r} F \eta_r}{RT} \right) - \exp\left(\frac{-\alpha_{c,r} F \eta_r}{RT} \right) \right] \tag{7.117}$$

or by its alternative form using Tafel slopes:

$$i_{F,r}=i_{o,r}\,exp\left[-\frac{E}{R}\left(\frac{1}{T}-\frac{1}{T_{ref}}\right)\right]\prod_{k=1}^{N}\left(\frac{C_{k,r}}{C_{k,r}^{ref}}\right)^{\nu_{K,r}}\left[exp\left(\frac{2.303}{\beta_{a,r}}\eta_r\right)-exp\left(-\frac{2.303}{\beta_{c,r}}\eta_r\right)\right] \tag{7.118}$$

where:

$i_{F,r}$ = Faradaic current density (A/m^2)

$i_{o,r}$ = exchange current density (A/m^2)

N = total number of chemical species

E = activation energy (J/kmol)

R = Universal Gas Constant (J/kmolK)

T = Temperature (K)

T_{ref} = reference temperature (K)

$C_{k,r}$ = molar concentration of species k in reaction r

$C_{k,r}^{ref}$ = reference molar concentration of species k in reaction r

$\alpha_{a,r}$ = anodic charge transfer coefficient

$\alpha_{c,r}$ = cathodic charge transfer coefficient

$\beta_{a,r}$ = Tafel slope for anodic reaction (V)

$\beta_{c,r}$ = Tafel slope for cathodic reaction (V)

F = Faraday constant (C/kmol)

η_r = overpotential (V)

Note:

Note that in Equation 7.117 (p. 267), $\alpha_{a,r}$ and $\alpha_{c,r}$ are charge transfer coefficients and not symmetry factors that can be often found in the literature for the Butler-Volmer equations.

The overpotential η_r is calculated as:

$$\eta_r=\varphi_{ed}-\varphi_{el}-E_{eq,r} \tag{7.119}$$

where $\varphi_{ed}$and φ_{el} are the electrode and electrolyte potentials (V), respectively, and $E_{eq,r}$ is the equilibrium potential (V).

The general reaction shown in Equation 7.116 (p. 267) occurs in both forward and backward directions simultaneously. The anodic reaction occurs in the direction where electrons are produced, while the cathodic reaction occurs in the direction where electrons are consumed.

The first term in Equation 7.117 (p. 267) is the rate of the anodic direction, while the second term is the rate of the cathodic direction. The difference between these two rates gives the net rate of reactions. The net direction of reaction depends on the sign of the surface overpotential. If overpotential is positive, the overall reaction moves in the direction of the anodic reaction, and if it is negative, it moves in the direction of the cathodic reaction.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

The general reaction (Equation 7.116 (p. 267)) also shows that electric current is proportional to the species production/consumption rate in an electrochemical reaction. This correlation for the species i in the r^{th} reaction is expressed by Faraday's law:

$$S^r_{Faradaic,i} = \frac{(\nu''_{i,r} - \nu'_{i,r}) M_{W,i}}{n_r F} i_{F,r} \tag{7.120}$$

where $S^r_{Faradaic,i}$ is the species production/consumption rate, $M_{W,i}$ is the species molecular weight, and n_r is the total number of electrons produced by the electrochemical reaction r calculated as:

$$n_r = \sum_k \left(\nu''_{k,r} - \nu'_{k,r}\right) z_k \tag{7.121}$$

The total species production/consumption rate for specie i due to all electrochemical reactions can be computed as:

$$S_{Faradaic,i} = \sum_r S^r_{Faradaic,i} \tag{7.122}$$

As follows from Faraday's law (Equation 7.120 (p. 269)), the rate of an electrochemical reaction can be expressed in terms of either electric current or species mass change, since these two quantities are directly proportional.

According to the Butler-Volmer equation (Equation 7.117 (p. 267)), the driving force for the electrochemical reaction is the potential difference across the electrode-electrolyte interface. The electric potential in the medium can be derived from the charge conservation law:

$$\nabla \cdot \mathbf{i} = 0 \tag{7.123}$$

where $\mathbf{i}$ is the electric current density vector.

In a solid phase, the current density and the gradient in potential are related by Ohm's law:

$$\mathbf{i} = -\sigma \nabla \varphi \tag{7.124}$$

where σ is the electric conductivity, and φ is the electric potential.

Therefore, the potential field is governed by a Laplace equation:

$$\nabla \cdot \left(\sigma \nabla \varphi\right) = 0 \tag{7.125}$$

In a liquid (or electrolyte) phase, electric current is the net flux of charged species:

$$\mathbf{i} = F \sum_k z_k \mathbf{N}_k = F \left(- \sum_k z_k D_k \nabla c_k + \mathbf{u} \sum_k z_k c_k - \nabla \varphi \sum_k z_k^2 m_k F c_k \right) \tag{7.126}$$

where:

$\mathbf{N}_k$ = flux density of species k generated by species diffusion, species convection, and the migration of species (solid or liquid) in the electric field

$\mathbf{u}$ = velocity of the flow field

c_k = concentration of species k

D_k = diffusion coefficient of species k

m_k = mobility of species k

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

In general, the treatment of electrochemical theory is too complicated to be practical. To simplify Equation 7.126 (p. 269), charge neutrality is usually assumed:

$$\sum_k z_k c_k = 0 \tag{7.127}$$

This eliminates the second term in Equation 7.126 (p. 269).

In addition, the first term in Equation 7.126 (p. 269) is usually considered negligible compared to the last term. This is a valid assumption for mixtures that are well-mixed or mixtures with a high electrolyte concentration. As a result, only the last term remains in Equation 7.126 (p. 269).

Defining the ionic conductivity as,

$$\sigma = F^2 \sum_k z_k^2 m_k c_k \tag{7.128}$$

the charge conservation equation (Equation 7.126 (p. 269)) reduces to:

$$\nabla \cdot \left(\sigma \nabla \varphi \right) = 0 \tag{7.129}$$

Therefore, the same Laplace equation is solved in both solid and liquid zones.

Charge neutrality is not explicitly enforced. Instead, it is assumed that there are unrepresented species of identical opposite charge that cancel the represented species charge.

The electric field may exert a force on the charged species in the electrolyte. This results in an additional term in the species transport equation (Equation 7.1 (p. 238)):

$$S_{migration,i} = \nabla \cdot \left(D_i \frac{z_i F}{RT} \rho Y_i \nabla \varphi \right) \tag{7.130}$$

Electrochemistry can contribute two distinct sources of heat to the energy equation (Equation 5.1 (p. 172)).

The first term is due to the charge-transfer reaction at the Faradaic interface and is modeled as:

$$\dot{q}_{Faradaic} = \sum_r \left| i_{F,r} \eta_r \right| \tag{7.131}$$

where $i_{F,r}$ is Faradaic current density, and η_r is the over-potential (see Equation 7.119 (p. 268)).

The second term is Joule heating due to motion of charges:

$$\dot{q}_{Joule} = \sigma \left| \nabla \varphi \right|^2 \tag{7.132}$$

7.1.5. Reacting Channel Model

This section provides theoretical background about the reacting channel model. Information can be found in the following sections:

7.1.5.1. Overview

7.1.5.2. Reacting Channel Model Theory

For more information about using the reacting channel model, see Reacting Channel Model in the User's Guide.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

7.1.5.1. Overview

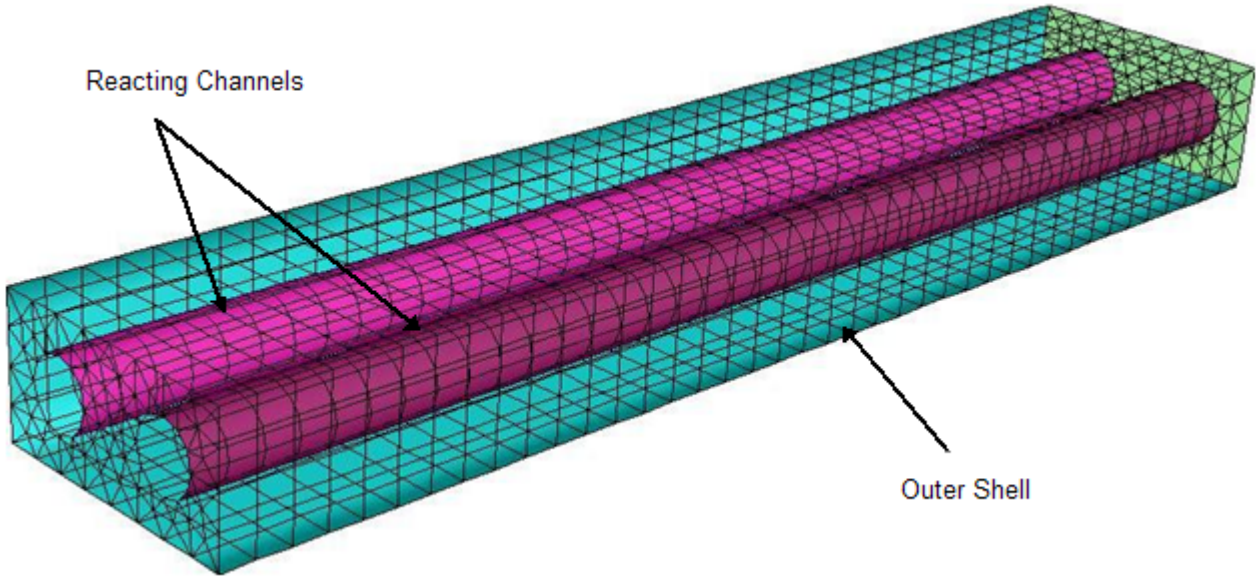
Some flow geometries contain long, thin channels (also called pipes or ducts) with reacting fluid inside the channels exchanging heat with the fluid outside the channels. Examples of such shell-and-tube configurations include fuel reformers and cracking furnaces. The flow inside the channels rapidly establishes a stationary, fully-developed radial profile. The evolution of chemical species and temperature in these channels can be accurately modeled with lower dimensional models, often assuming parabolic flow. This is much more efficient than resolving the mesh inside the channels and using a 3D solver for both flow and chemistry inside and outside the channels simultaneously. The reacting channel model inside Ansys Fluent allows affordable simulations of reactors with simple flow but complex chemistry occurring in long, thin channels, thermally coupled to geometrically complex outer flows with simple, or no, chemistry.

For the limitations that apply to the Reacting Channel model, see Overview and Limitations of the Reacting Channel Model in the *Fluent User's Guide*.

7.1.5.2. Reacting Channel Model Theory

In the reacting channel model, all channel walls must be geometrically resolved in three-dimensions. However, only the geometry in the outer shell should be meshed; no cells should be included inside the channel walls. Figure 7.2: Cross-section of a Channel and Outer Shell Around It. (p. 271) shows a typical configuration with two circular channels and an outer shell around it. The flow inside the channels is reacting, while the outer flow in the shell can be reacting or non-reacting. The channel walls are non-permeable and only heat transfer can take place through them.

Figure 7.2: Cross-section of a Channel and Outer Shell Around It.



During an Ansys Fluent iteration, the flow inside the reacting channel is solved using fixed wall temperatures from the 3D outer flow, and then the 3D outer flow is solved using fixed heat fluxes from the channel solver.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

7.1.5.2.1. Flow Inside the Reacting Channel

The flow inside the channels is assumed to be a plug flow, which means that the flow variables at any axial location are radially uniform. Axial diffusion of heat and species are also ignored and therefore change only by reaction. The plug flow evolution equations for the channel species and temperatures are

$$\frac{dY_k}{dt} = \frac{W_k \dot{\omega}_k}{\rho}$$
$$\frac{dT}{dt} = \frac{\sum_k W_k \dot{\omega}_k h_k + \dot{Q}_C}{\rho c_p} \tag{7.133}$$

where,

$Y_k = k^{th}$ species mass fraction

$W_k = k^{th}$ molecular weight

$\dot{\omega}_k = k^{th}$ net molar reaction rate

$h_k = k^{th}$ total enthalpy

ρ = mixture density

T = bulk temperature

c_p = specific heat

$\dot{Q}_C$ = convective heat transfer source

The reaction channel convective heat transfer source $\dot{Q}_C$ is determined as

$$\dot{Q}_C = h_f \frac{A_C}{V_C} \left(T_W - T \right) \tag{7.134}$$

where,

h_f = convective heat transfer coefficient on the channel side

A_c = wall area of the channel element

V_c = volume of the channel element

The channel wall temperature T_w (not to be mistaken with the channel bulk temperature T) is averaged from the 3D outer flow temperature field on the resolved channel wall, as described below.

The heat transfer coefficient h_f in Equation 7.134 (p. 272) is calculated as

$$h_f = \frac{Nu K_C}{D_C} \tag{7.135}$$

where K_c and D_c are the gas-phase thermal conductivity and the channel diameter, respectively. The Nusselt number, Nu, in the above equation is calculated from empirical correlations for fully developed pipe flow as follows:

- For Laminar flow (Re < 3000)

$$Nu = 3.66 \tag{7.136}$$

- For Turbulent flow (Re > 3000), Gnielinski's correlation for turbulent flow is used:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$Nu=\frac{f(\mathrm{Re}-1000)\mathrm{Pr}}{1+2.7\sqrt{f}\,(\mathrm{Pr}^{2/3}-1)}\left(\frac{T_c}{T_w}\right)^{0.36} \tag{7.137}$$

where f is the friction factor. For smooth tubes of circular cross-section, f is given by

$$f==\frac{0.125}{\left(0.79\log(\mathrm{Re})-1.64\right)^2} \tag{7.138}$$

The plug flow equations are solved with a stiff ODE solver using time-steps based on the grid size (size of channel element) and the local channel velocity.

7.1.5.2.2. Surface Reactions in the Reacting Channel

When the surface reaction option is enabled, additional source terms due to surface reactions are added to Equation 7.133 (p. 272). Details of the heat and mass sources due to surface reactions are described in Wall Surface Reactions and Chemical Vapor Deposition (p. 256). The reacting surface area of a reacting channel element is calculated as

$$A_{surf}=V_C\left(A/V\right)_{Ratio} \tag{7.139}$$

where V_C is the volume of the channel element and $\left(A/V\right)_{Ratio}$ is the area to volume ratio of the catalytic surface.

Note:

In the reacting channel model, the heat source due to surface reaction and the heat and mass source due to mass deposition are always included.

7.1.5.2.3. Porous Medium Inside Reacting Channel

The porous medium option in the reacting channel model uses the superficial velocity porous formulation. Ansys Fluent calculates the superficial mixture velocities based on the volumetric flow rate in a porous region.

When the **Porous Medium** option is enabled, the heat transfer coefficient calculated from Equation 7.135 (p. 272) is modified to account for the solid material of the porous medium in the following manner.

$$h_f=\alpha\frac{NuK_C}{D_C}+\left(1-\alpha\right)\frac{K_{S,C}}{D_C} \tag{7.140}$$

where α is the porosity and $K_{S,C}$ is the conductivity of the solid material. Also, the reaction rates $\dot{\omega}_k$ are modified by multiplying $\dot{\omega}_k$ by medium porosity α to account for the displacement of the gas by the solid.

The **Porous Medium** option provides the capability to model pressure drop inside the reacting channel. The pressure drop in the channel is calculated only in the axial direction and has contributions from viscous and inertial resistances:

$$\Delta p=\Delta p_{Viscous}+\Delta p_{Inertial} \tag{7.141}$$

The pressure drop due to viscous resistance is calculated using Darcy's law

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$\Delta p_{Viscous} = C_{Viscous}\,\mu\, u\, \Delta x \tag{7.142}$$

where $C_{Viscous}$ is the viscous resistance along axial direction and Δx is the length of the channel element.

The pressure drop due to inertial losses is modeled as

$$\Delta p_{Inertial} = \frac{\rho}{2} C_{Inertial}\, u^2\, \Delta x \tag{7.143}$$

where $C_{Inertial}$ is the inertial resistance along axial direction and Δx is the length of the channel element.

7.1.5.2.4. Outer Flow in the Shell

The outer shell can be any arbitrarily shaped 3D geometry, but must resolve the channel walls. Note that while this outer shell is meshed in Ansys Fluent with finite-volume cells, the inside of the channels should not be meshed. This outer flow can be a reacting or non-reacting mixture, which is typically a different mixture material than the complex mechanism material used for the reacting channel plug flow.

The Ansys Fluent energy solution of the outer shell flow uses a prescribed heat flux boundary condition at the channel walls from the solution of the reacting channel. The value of this heat flux is calculated as an under-relaxation of the heat gained from the channel:

$$\dot{Q}_{3D} = -\alpha \dot{Q}_C - (1-\alpha)\dot{Q}_C^o \tag{7.144}$$

where $\dot{Q}_C$ is the channel heat gain/loss, α is a user-specified under-relaxation parameter, and $\dot{Q}_C^o$ is the heat flux from the previous iteration. At convergence, the heat loss/gain from the channel is equal to the heat gain/loss in the shell outer flow.

7.1.6. Reactor Network Model

The reactor network model is used to simulate species and temperature fields in a combustor using a detailed chemical kinetic mechanism. The reactor network is constructed from a converged Ansys Fluent simulation modeled with a fast chemistry combustion model, such as the Non-Premixed, Partially-Premixed, or Eddy-Dissipation model. A full chemical mechanism in CHEMKIN format can be imported into Fluent and solved on the reactor network. The combustor volume is automatically subdivided into a small number of connected perfectly stirred reactors. The mass fluxes through the network are determined from the CFD solution, and the species and temperatures in the reactor network are solved together. Hence, the reactor network model is used to simulate finite-rate chemistry effects with detailed kinetic mechanisms, in particular pollutant emissions such as NOx, CO, and unburnt hydro-carbons. Since the specified number of stirred reactors is much smaller than the number of CFD cells, the reactor network model allows much faster simulations of the species and temperature fields than solving for detailed chemistry in every cell, as in the Laminar, EDC, and PDF Transport models.

Typically, the reactor network model is executed on a converged steady-state RANS solution or a time-averaged unsteady solution. The model can also be run on an unsteady flow representing a "snap-shot" in time. Since there is no backward coupling of the reactor-network solution to the flow, the model is useful for predictions where the detailed chemistry has little impact on the flow, such as pollutant formation. The model is inappropriate for highly unsteady flows such as flame ignition

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

or global extinction and also for flows that are strongly influenced by the chemistry, such as soot with significant soot-radiation interaction.

For more information about using reactor network, see Reactor Network Model in the *Fluent User's Guide*.

7.1.6.1. Reactor Network Model Theory

The first step in a reactor network simulation involves agglomerating the CFD cells into the specified number of reactors, $N_{reactor}$.

Since each reactor is a perfectly mixed representation of a region of the combustor, ideally cells that are closest in composition space should be grouped together. For optimal performance, the CFD cells grouped together in each reactor should have temperatures and species mass fractions that are as similar as possible.

By default, for Non-Premixed and Partially-Premixed cases, Fluent groups cells that have similar temperatures and mixture fractions, and for Species Transport cases, Fluent groups cells that have similar temperatures and mass fractions of N_2 and H_2O. These defaults should provide good cell clustering in most cases. However, Fluent allows user-controlled clustering through custom-field functions when these defaults are not sufficient.

After similar cells have been clustered, Fluent splits non-contiguous groups, then agglomerates clusters with the smallest number of cells to their closest neighbors until the specified number of reactors, $N_{reactor}$, is obtained.

The second step in a reactor network simulation involves the solution of the reactor network, which proceeds as follows.

The mass flux matrix, $\dot{m}$, is calculated from the cell fluxes in the CFD solution, where each matrix component $\dot{m}_{ij}$ is the mass flux from reactor i to reactor j. The k^{th} species mass fraction in reactor i, Y_k^i, is governed by the algebraic equation:

$$\sum_{j=1,j\neq i}^{N_{reactor}} \dot{m}_{ij} Y_k^j - \dot{m}_i Y_k^i = V_i \dot{\omega}_k^i + S_k^i \tag{7.145}$$

where V_i is the volume of reactor i, $\dot{\omega}_k^i$ is the k^{th} species reaction rate in the i^{th} reactor, and S_k^i is a mass source term. $\dot{m}_i$ is the net mass flux into reactor i and is calculated as:

$$\dot{m}_i = \sum_{j=1,j\neq i}^{N_{reactor}} \dot{m}_{ij} \tag{7.146}$$

The mass source term, S_k^i, accounts for contributions from the CFD inlet species mass fluxes through the CFD boundaries and from volume sources, such as the Discrete Phase Model (DPM).

The system of equations for Y_k^i, Equation 7.145 (p. 275), has the dimension of $N_{reactor} \times N_{species}$, where $N_{reactor}$ is the user-specified number of reactors, and $N_{species}$ is the number of species in the chemical mechanism. Ansys Fluent solves this system with a segregated algorithm by default, although the option to use a fully coupled algorithm is available.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

7.1.6.1.1. Reactor network temperature solution

An energy equation is not solved in the reactor network. Instead, by default, the temperature in each reactor is calculated from the equation of state. The reactor pressure is fixed and determined as the mass-averaged pressure of the CFD cells in the reactor. Note that the density of each reactor is fixed since both the reactor volume and the reactor mass are constant. This approach ensures that heat loss (or gain) in the CFD simulation is appropriately accounted for in the reactor network. Fluent also provides an option to not solve for temperature, in which case the temperature is fixed as the mass-average temperature of the CFD cells in the reactor.

7.2. Composition PDF Transport

Ansys Fluent provides a composition PDF transport model for modeling finite-rate chemistry effects in turbulent flames. For information about using the composition PDF transport model, see Modeling a Composition PDF Transport Problem in the User's Guide. Information about the theory behind this model is presented in the following sections:

7.2.1. Overview

7.2.2. Composition PDF Transport Theory

7.2.3. The Lagrangian Solution Method

7.2.4. The Eulerian Solution Method

7.2.1. Overview

The composition PDF transport model, like the Laminar Finite-Rate (see Direct Use of Finite-Rate Kinetics (no TCI) (p. 243)) and EDC model (see The Eddy-Dissipation-Concept (EDC) Model (p. 249)), should be used when you are interested in simulating finite-rate chemical kinetic effects in turbulent reacting flows. With an appropriate chemical mechanism, kinetically-controlled species such as CO and NOx, as well as flame extinction and ignition, can be predicted. PDF transport simulations are computationally expensive, and it is recommended that you start your modeling with small meshes, and preferably in 2D.

Ansys Fluent has two different discretizations of the composition PDF transport equation, namely Lagrangian and Eulerian. The Lagrangian method is strictly more accurate than the Eulerian method, but requires significantly longer run time to converge.

For limitations that exist with the composition PDF transport mode in Ansys Fluent, see Limitation in the *Fluent User's Guide*.

7.2.2. Composition PDF Transport Theory

Turbulent combustion is governed by the reacting Navier-Stokes equations. While this equation set is accurate, its direct solution (where all turbulent scales are resolved) is far too expensive for practical turbulent flows. In Species Transport and Finite-Rate Chemistry (p. 237), the species equations are Reynolds-averaged, which leads to unknown terms for the turbulent scalar flux and the mean reaction rate. The turbulent scalar flux is modeled in Ansys Fluent by gradient diffusion, treating turbulent convection as enhanced diffusion. The mean reaction rate can be modeled with the Laminar, Eddy-Dissipation or EDC Finite-Rate chemistry models. Since the reaction rate is invariably highly nonlinear, modeling the mean reaction rate in a turbulent flow is difficult and prone to error.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

An alternative to Reynolds-averaging the species and energy equations is to derive a transport equation for their single-point, joint probability density function (PDF). This PDF, denoted by P, can be considered to represent the fraction of the time that the fluid spends at each species and temperature state. P has $N+1$ dimensions for the N species and temperature spaces. From the PDF, any single-point thermo-chemical moment (for example, mean or RMS temperature, mean reaction rate) can be calculated. The composition PDF transport equation is derived from the Navier-Stokes equations as [525] (p. 1083):

$$\begin{aligned}\frac{\partial}{\partial t}\left(\rho P\right)+\frac{\partial}{\partial x_i}\left(\rho u_i P\right)+\frac{\partial}{\partial \psi_k}\left(\rho S_k P\right)= & -\frac{\partial}{\partial x_i}\left[\rho\langle u_i''|\psi\rangle P\right] \\ & +\frac{\partial}{\partial \psi_k}\left[\rho\left\langle \frac{1}{\rho}\frac{\partial J_{i,k}}{\partial x_i}\middle|\psi\right\rangle P\right]\end{aligned} \tag{7.147}$$

where

P = Favre joint PDF of composition

ρ = mean fluid density

u_i = Favre mean fluid velocity vector

S_k = reaction rate for species k

ψ = composition space vector

u_i'' = fluid velocity fluctuation vector

$J_{i,k}$ = molecular diffusion flux vector

The notation of $\langle ... \rangle$ denotes expectations, and $\langle A|B\rangle$ is the conditional probability of event A, given that event B occurs.

In Equation 7.147 (p. 277), the terms on the left-hand side are closed, while those on the right-hand side are not and require modeling. The first term on the left-hand side is the unsteady rate of change of the PDF, the second term is the change of the PDF due to convection by the mean velocity field, and the third term is the change due to chemical reactions. The principal strength of the PDF transport approach is that the highly-nonlinear reaction term is completely closed and requires no modeling. The two terms on the right-hand side represent the PDF change due to scalar convection by turbulence (turbulent scalar flux), and molecular mixing/diffusion, respectively.

The turbulent scalar flux term is unclosed, and is modeled in Ansys Fluent by the gradient-diffusion assumption

$$-\frac{\partial}{\partial x_i}\left[\rho\langle u_i''|\psi\rangle P\right]=\frac{\partial}{\partial x_i}\left(\frac{\rho\mu_t}{Sc_t}\frac{\partial P}{\partial x_i}\right) \tag{7.148}$$

where μ_t is the turbulent viscosity and Sc_t is the turbulent Schmidt number. A turbulence model, as described in Turbulence (p. 41), is required for composition PDF transport simulations, and this determines μ_t.

Since single-point PDFs are described, information about neighboring points is missing and all gradient terms, such as molecular mixing, are unclosed and must be modeled. The mixing model is critical because combustion occurs at the smallest molecular scales when reactants and heat diffuse together. Modeling mixing in PDF methods is not straightforward, and is the weakest link in the PDF transport approach. See Particle Mixing (p. 279) for a description of the mixing models.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

7.2.3. The Lagrangian Solution Method

A Lagrangian Monte Carlo method is used to solve for the $N+1$ dimensional PDF Transport equation. Monte Carlo methods are appropriate for high-dimensional equations since the computational cost increases linearly with the number of dimensions. The disadvantage is that statistical errors are introduced, and these must be carefully controlled.

To solve the modeled PDF transport equation, an analogy is made with a stochastic differential equation (SDE) that has identical solutions. The Monte Carlo algorithm involves notional particles that move randomly through physical space due to particle convection, and also through composition space due to molecular mixing and reaction. The particles have mass and, on average, the sum of the particle masses in a cell equals the cell mass (cell density times cell volume). Since practical meshes have large changes in cell volumes, the particle masses are adjusted so that the number of particles in a cell is controlled to be approximately constant and uniform.

The processes of convection, diffusion, and reaction are treated in fractional steps as described in the sections that follow. For information on the fractional step method, refer to [92] (p. 1058).

Information about this method is described in the following sections:

7.2.3.1. Particle Convection

7.2.3.2. Particle Mixing

7.2.3.3. Particle Reaction

7.2.3.1. Particle Convection

A spatially second-order-accurate Lagrangian method is used in Ansys Fluent, consisting of two steps. At the first convection step, particles are advanced to a new position

$$x_i^{1/2} = x_i^0 + \frac{1}{2} u_i^0 \Delta t \tag{7.149}$$

where

x_i = particle position vector

u_i = Favre mean fluid-velocity vector at the particle position

Δt = particle time step

For unsteady flows, the particle time step is the physical time step. For steady-state flows, local time steps are calculated for each cell as

$$\Delta t = \min\left(\Delta t_{conv}, \Delta t_{diff}, \Delta t_{mix}\right) \tag{7.150}$$

where

Δt_{conv} = *convection number* $\times \Delta x$ / (*cell fluid velocity*)

Δt_{diff} = *diffusion number* $\times \left(\Delta x\right)^2$ / (*cell turbulent diffusivity*)

Δt_{mix} = *mixing number* $\times$ *turbulent time scale*

Δx = characteristic cell length = $volume^{1/D}$ where D is the problem dimension

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

After the first convection step, all other sub-processes, including diffusion and reaction are treated. Finally, the second convection step is calculated as

$$x_i^1 = x_i^{1/2} + \Delta t\left(u_i^{1/2} - \frac{1}{2}u_i^0 + \frac{1}{\rho Sc_t}\frac{\partial \mu_{eff}}{\partial x_i} + \xi_i\sqrt{\frac{2\mu_{eff}}{\rho \Delta t Sc_t}} \right) \tag{7.151}$$

where

ρ = mean cell fluid density

u_i = mean fluid-velocity vector at the particle position

μ_{eff} = effective viscosity

Sc_t = turbulent Schmidt number

ξ_i = standardized normal random vector

7.2.3.2. Particle Mixing

Molecular mixing of species and heat must be modeled and is usually the source of the largest modeling error in the PDF transport approach. Ansys Fluent provides three models for molecular diffusion: the Modified Curl model [277] (p. 1069), [477] (p. 1080), the IEM model (which is sometimes called the LSME model) [146] (p. 1061) and the EMST model [632] (p. 1089).

7.2.3.2.1. The Modified Curl Model

For the Modified Curl model, a few particle pairs are selected at random from all the particles in a cell, and their individual compositions are moved toward their mean composition. For the special case of equal particle mass, the number of particle pairs selected is calculated as

$$N_{pair} = \frac{1.5 C_\varphi N \Delta t}{\tau_t} \tag{7.152}$$

where

N = total number of particles in the cell

C_φ = mixing constant (default = 2)

τ_t = turbulent time scale (for the k-ε model this is k/ε)

The algorithm in [477] (p. 1080) is used for the general case of variable particle mass.

For each particle pair, a uniform random number ξ is selected and each particle's composition φ is moved toward the pair's mean composition by a factor proportional to ξ:

$$\begin{aligned} \varphi_i^1 &= (1-\xi)\varphi_i^0 + \xi\frac{\left(\varphi_i^0 m_i + \varphi_j^0 m_j\right)}{\left(m_i + m_j\right)} \\ \varphi_j^1 &= (1-\xi)\varphi_j^0 + \xi\frac{\left(\varphi_i^0 m_i + \varphi_j^0 m_j\right)}{\left(m_i + m_j\right)} \end{aligned} \tag{7.153}$$

where φ_i and φ_j are the composition vectors of particles i and j, and m_i and m_j are the masses of particles i and j.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

7.2.3.2.2. The IEM Model

For the Interaction by Exchange with the Mean (IEM) model, the composition of all particles in a cell are moved a small distance toward the mean composition:

$$\varphi^1 = \varphi^0 - \left(1 - e^{-0.5C_\varphi \Delta t/\tau_t}\right)\left(\varphi^0 - \widetilde{\varphi}\right) \tag{7.154}$$

where φ^0 is the composition before mixing, φ^1 is the composition after mixing, and $\widetilde{\varphi}$ is the Favre mean-composition vector at the particle's location.

7.2.3.2.3. The EMST Model

Physically, mixing occurs between fluid particles that are adjacent to each other. The Modified Curl and IEM mixing models take no account of this localness, which can be a source of error. The Euclidean Minimum Spanning Tree (EMST) model mixes particle pairs that are close to each other in composition space. Since scalar fields are locally smooth, particles that are close in composition space are likely to be close in physical space. The particle pairing is determined by a Euclidean Minimum Spanning Tree, which is the minimum length of the set of edges connecting one particle to at least one other particle. The EMST mixing model is more accurate than the Modified Curl and IEM mixing models, but incurs a slightly greater computational expense. Details on the EMST model can be found in reference [632] (p. 1089).

7.2.3.2.4. Liquid Reactions

Reactions in liquids often occur at low turbulence levels (small Re), among reactants with low diffusivities (large Sc). For such flows, the mixing constant default of C_φ=2 overestimates the mixing rate. The **Liquid Micro-Mixing** option interpolates C_φ from model turbulence [527] (p. 1083) and scalar [187] (p. 1063) spectra.

7.2.3.3. Particle Reaction

The particle composition vector is represented as

$$\varphi = \left(Y_1, Y_2, ..., Y_N, T\right) \tag{7.155}$$

where Y_k is the k^{th} species mass fraction, and T is the temperature.

For the reaction fractional step, the reaction source term is integrated as

$$\varphi^1 = \varphi^0 + \int_0^{\Delta t} S dt \tag{7.156}$$

where S is the chemical source term. Most realistic chemical mechanisms consist of tens of species and hundreds of reactions. Typically, a reaction does not occur until an ignition temperature is reached, but then proceeds very quickly until reactants are consumed. Hence, some reactions have very fast time scales, in the order of 10^{-10} s, while others have much slower time scales, on the order of 1 s. This time-scale disparity results in numerical stiffness, which means that extensive computational work is required to integrate the chemical source term in Equation 7.156 (p. 280). In Ansys Fluent, the reaction step (that is, the calculation of φ^1) can be performed either by Direct Integration or by In-Situ Adaptive Tabulation (ISAT), as described in the following paragraphs.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

A typical steady-state PDF transport simulation in Ansys Fluent may have 50000 cells, with 20 particles per cell, and requires 1000 iterations to converge. Hence, at least 10^9 stiff ODE integrations are required. Since each integration typically takes tens or hundreds of milliseconds, on average, the direct integration of the chemistry is extremely CPU-demanding.

For a given reaction mechanism, Equation 7.156 (p. 280) may be considered as a mapping. With an initial composition vector φ^0, the final state φ^1 depends only on φ^0 and the mapping time Δt. In theory, if a table could be built before the simulation, covering all realizable φ^0 states and time steps, the integrations could be avoided by table look-ups. In practice, this *a priori* tabulation is not feasible since a full table in $N+3$ dimensions (N species, temperature, pressure and time step) is required. To illustrate this, consider a structured table with M points in each dimension. The required table size is M^{N+3}, and for a conservative estimate of $M=10$ discretization points and $N=7$ species, the table would contain 10^{10} entries.

On closer examination, the full storage of the entire realizable space is very wasteful because most regions are never accessed. For example, it would be unrealistic to find a composition of $Y_{OH}=1$ and $T=300K$ in a real combustor. In fact, for steady-state, 3D laminar simulations, the chemistry can be parameterized by the spatial position vector. Thus, mappings must lie on a three dimensional manifold within the $N+3$ dimensional composition space. It is, hence, sufficient to tabulate only this accessed region of the composition space.

The accessed region, however, depends on the particular chemical mechanism, molecular transport properties, flow geometry, and boundary conditions. For this reason, the accessed region is not known before the simulation and the table cannot be preprocessed. Instead, the table must be built during the simulation, and this is referred to as *in-situ* tabulation. Ansys Fluent employs ISAT [523] (p. 1083) to dynamically tabulate the chemistry mappings and accelerate the time to solution. ISAT is a method to tabulate the accessed composition space region "on the fly" (in-situ) with error control (adaptive tabulation). When ISAT is used correctly, accelerations of two to three orders of magnitude are typical. However, it is important to understand how ISAT works in order to use it optimally.

7.2.4. The Eulerian Solution Method

The Lagrangian solution method solves the composition PDF transport equation by stochastically tracking Lagrangian particles through the domain. It is computationally expensive since a large number of particles are required to represent the PDF, and a large number of iterations are necessary to reduce statistical errors and explicitly convect the particles through the domain. The Eulerian PDF transport model overcomes these limitations by assuming a shape for the PDF, which allows Eulerian transport equations to be derived. Stochastic errors are eliminated and the transport equations are solved implicitly, which is computationally economical. The multi-dimensional PDF shape is assumed as a product of delta functions. As with the Lagrangian PDF model, the highly nonlinear chemical source term is closed. However, the turbulent scalar flux and molecular mixing terms must be modeled, and are closed with the gradient diffusion and the IEM models, respectively.

The composition PDF of N_s+1 dimension (N_s species and enthalpy) is represented as a collection of N_e delta functions (or modes). This presumed PDF has the following form:

$$P(\psi;\vec{x},t)=\sum_{n=1}^{N_e} p_n(\vec{x},t)\prod_{k=1}^{N_s}\delta\left[\psi_k-<\varphi_k>_n(\vec{x},t)\right] \tag{7.157}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

where p_n is the probability in each mode, $<\varphi_k>_n$ is the conditional mean composition of species k in the n^{th} mode, ψ_k is the composition space variable of species k, and $\delta(...)$ is the delta function.

The Eulerian PDF transport equations are derived by substituting Equation 7.157 (p. 281) into the closed composition PDF transport equation (Equation 7.147 (p. 277) with Equation 7.148 (p. 277) and Equation 7.154 (p. 280)). The unknown terms, p_n and $<\varphi_k>_n$, are determined by forcing lower moments of this transported PDF to match the RANS lower moment transport equations, using the Direct Quadrature Method of Moments (DQMOM) approach [187] (p. 1063), [409] (p. 1076). The resulting transport equations are:

- Probability (magnitude of the n^{th} delta function):

$$\frac{\partial \rho p_n}{\partial t}+\frac{\partial}{\partial x_i}\left(\rho u_i p_n\right)=\nabla\left(\rho \Gamma \nabla p_n\right) \tag{7.158}$$

- Probability weighted conditional mean of composition k:

$$\frac{\partial \rho s_{k,n}}{\partial t}+\frac{\partial}{\partial x_i}\left(\rho u_i s_{k,n}\right)=\nabla\left(\rho \Gamma \nabla s_{k,n}\right)+\rho\left(M_{k,n}+S_{k,n}+C_{k,n}\right) \tag{7.159}$$

where p_n is the probability of the n^{th} mode, and $s_{k,n}=p_n<\varphi_k>_n$ is the k^{th} species probability weighted conditional mean composition of the n^{th} mode. $\Gamma=\mu_l+\mu_t / Sc_t$ is the effective turbulent diffusivity. The terms $M_{k,n}$, $S_{k,n}$ and $C_{k,n}$ represent mixing, reaction and correction terms respectively. Note that only N_e-1 probability equations are solved and the N^{th} probability is calculated as one minus the sum of the N_e-1 solved probabilities.

For more information, see the following sections:

7.2.4.1. Reaction
7.2.4.2. Mixing
7.2.4.3. Correction
7.2.4.4. Calculation of Composition Mean and Variance

7.2.4.1. Reaction

The reaction source term $S_{k,n}$ in Equation 7.159 (p. 282) for the k^{th} composition and the n^{th} mode is calculated as,

$$S_{k,n}=p_n S\left(<\varphi_k>_n\right)_k \tag{7.160}$$

where $S(\,)_k$ is the net reaction rate for the k^{th} component.

7.2.4.2. Mixing

The micro-mixing term $M_{k,n}$ is modeled with the IEM mixing model:

$$M_{k,n}=\frac{C_\varphi}{\tau}\left(<\varphi_k>_n-\psi_k\right) \tag{7.161}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

where τ is the turbulence time-scale and C_φ is the mixing constant.

Hence, for the two-mode DQMOM-IEM model, the mixing terms for component k are,

$$\begin{aligned} M_{k,1} &= \frac{C_\varphi}{\tau}\left(p_1 s_{k,2} - p_2 s_{k,1}\right) \\ M_{k,2} &= \frac{C_\varphi}{\tau}\left(p_2 s_{k,1} - p_1 s_{k,2}\right) \end{aligned} \tag{7.162}$$

The default value of C_φ is 2, which is appropriate for gas-phase combustion. For reactions in liquids, where the diffusivities are much smaller than gases, the **Liquid Micro-Mixing** option interpolates C_φ from model turbulence [527] (p. 1083) and scalar [187] (p. 1063) spectra.

7.2.4.3. Correction

Using assumptions to ensure realizability and boundedness, the correction terms $C_{k,n}$ in Equation 7.159 (p. 282) for the k^{th} composition are determined from the linear system,

$$\sum_{n=1}^{N_e} <\varphi_k>_n^{m_k-1} C_{k,n} = \sum_{n=1}^{N_e} \left(m_k - 1\right) <\varphi_k>_n^{m_k-2} p_n c_{k,n} \tag{7.163}$$

where m_k are the non-negative integer lower moments (1–N_e) for each component k. Note that the condition of the matrix decreases with increasing m_k, which reduces the stability of higher mode simulations.

The dissipation term $c_{k,n}$ in Equation 7.163 (p. 283) is calculated as,

$$c_{k,n} = \Gamma \nabla\left(<\varphi_k>_n\right) \cdot \nabla\left(<\varphi_k>_n\right) \tag{7.164}$$

For the two-mode DQMOM-IEM model, the correction terms for the k^{th} component are,

$$\begin{aligned} C_{k,1} &= \frac{\Gamma}{<\varphi_k>_1 - <\varphi_k>_2}\left(p_1\left(\nabla <\varphi_k>_1\right)^2 + p_2\left(\nabla <\varphi_k>_2\right)^2\right) \\ C_{k,2} &= -C_{k,1} \end{aligned} \tag{7.165}$$

7.2.4.4. Calculation of Composition Mean and Variance

The mean composition (species k or energy) is calculated as,

$$\overline{\varphi_k} = \sum_{n=1}^{N_e} s_{k,n} \tag{7.166}$$

and its variance is calculated as

$$\overline{\varphi_k'^2} = \left(\sum_{n=1}^{N_e} s_{k,n} <\varphi_k>_n\right) - \overline{\varphi}_k^{\,2} \tag{7.167}$$

7.3. Chemistry Acceleration

Ansys Fluent provides several methods to reduce the computational expense of solving detailed chemistry in multi-dimensional CFD. For information about using these chemistry acceleration methods,

see Using Chemistry Acceleration in the User's Guide. Information about the theory behind these approaches is presented in the following sections:

7.3.1. Overview and Limitations
7.3.2. In-Situ Adaptive Tabulation (ISAT)
7.3.3. Dynamic Mechanism Reduction
7.3.4. Chemistry Agglomeration
7.3.5. Chemical Mechanism Dimension Reduction
7.3.6. Dynamic Cell Clustering with Ansys Fluent CHEMKIN-CFD Solver
7.3.7. Dynamic Adaptive Chemistry with Ansys Fluent CHEMKIN-CFD Solver

7.3.1. Overview and Limitations

Ansys Fluent has the following turbulence-chemistry interaction options for simulating combustors with detailed chemical kinetics:

- Finite-Rate chemistry with no turbulence-chemistry interaction model (for laminar flows or for turbulent flows where turbulence-chemistry interaction may be neglected)
- Eddy-Dissipation Concept (for turbulent flames)
- Lagrangian Composition PDF Transport (for turbulent flames)
- Eulerian Composition PDF Transport (for turbulent flames)

Using detailed chemistry is appropriate when modeling kinetically controlled phenomena, such as slowly forming product and pollutant species, as well as flame ignition and extinction. Comprehensive chemical mechanisms contain a multitude of intermediate species in addition to the major fuel, oxidizer, and product species. These intermediate species evolve at widely different reaction rates resulting in disparate species formation and destruction time-scales. The numerical time integration to accurately calculate the species evolution requires very small time substeps, called stiffness, and causes large computational run times. The chemistry acceleration tools in Ansys Fluent can mitigate this cost, but with some loss of accuracy. Care must be taken to set controlling parameters so that these inaccuracies are within acceptable limits.

7.3.2. In-Situ Adaptive Tabulation (ISAT)

ISAT is Ansys Fluent's most powerful tool to accelerate detailed stiff chemistry. A speedup of two to three orders of magnitude is common, which is significant in that a simulation that would take months without ISAT can be run in days instead.

For kinetic mechanisms that are deterministic, the final reacted state is a unique function of the initial unreacted state and the time step. This reaction mapping can, in theory, be performed once and tabulated. The table can then be interpolated with run-time speedup as long as interpolation is more efficient than chemistry integration. In practice, pre-tabulation is prohibitive since the table dimensions are too large: the number of species (N), plus the temperature, pressure, and time step. However, in a reacting flow, only a very small sub-space of the full composition space is accessed. For example, it is unlikely that a state consisting of pure radical OH (mass fraction of one) at a temperature of 300 K will exist, and it is unnecessary to pre-tabulate this state. The chemistry is said to lie on a low-dimensional manifold. In fact, for steady-state, 3D, laminar simulations, the chemistry can be paramet-

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

erized by the spatial position vector. Thus, mappings must lie on a three dimensional manifold within the N+3 dimensional composition space. It is, hence, sufficient to tabulate only this accessed region of the composition space. The accessed region, however, depends on the particular chemical mechanism, the thermodynamic and transport properties, the flow geometry, and the boundary conditions. For this reason, the accessed region is not known before the simulation and the table cannot be preprocessed. Instead, the table must be built during the simulation, and this is referred to as in-situ tabulation. Ansys Fluent employs ISAT [523] (p. 1083) to dynamically tabulate the chemistry mappings and accelerate the time to solution. ISAT is a method to tabulate the accessed composition space region "on the fly" (in-situ) with error control (adaptive tabulation). It is important to understand how ISAT works in order to use it optimally.

Reaction over a time step Δt from an initial composition ϕ^0 to a final composition ϕ^1 is calculated as:

$$\phi^1 = \phi^0 + \int_0^{\Delta t} S dt \tag{7.168}$$

where S is the chemical source term.

At the start of an Ansys Fluent simulation using ISAT, the ISAT table is empty. For the first reaction step, Equation 7.168 (p. 285) is integrated with a stiff ODE solver. This is called Direct Integration (DI). The first table entry is created and consists of:

- the initial composition ϕ^0 (where the superscript 0 denotes the composition vector *before* the reaction)
- the mapping ϕ^1 (where the superscript 1 denotes the composition vector *after* the reaction)
- the mapping gradient matrix $A = \partial\phi^1 / \partial\phi^0$
- a hyper-ellipsoid of accuracy

The next reaction mapping is calculated as follows: The initial composition vector for this state is denoted by ϕ_q^0, where the subscript q denotes a *query*. The existing table (consisting of one entry at this stage) is queried by interpolating the new mapping as

$$\phi_q^1 = \phi^1 + A\left(\phi_q^0 - \phi^0\right) \tag{7.169}$$

The mapping gradient is hence used to linearly interpolate the table when queried. The ellipsoid of accuracy (EOA) is the elliptical space around the table point ϕ^0 where the linear approximation to the mapping is accurate to the specified tolerance, ε_{tol}.

If the query point ϕ_q^0 is within the EOA, then the linear interpolation by Equation 7.169 (p. 285) is sufficiently accurate, and the mapping is *retrieved*. Otherwise, a direct integration (DI) is performed and the mapping error $\varepsilon = \left| B\left(\phi_{DI}^1 - \phi_q^1\right)\right|$ is calculated (here, B is a scaling matrix). If this error is smaller than the specified error tolerance (ε_{tol}), then the original interpolation ϕ_q^1 is accurate and the EOA is *grown* so as to include ϕ_q^0. If not, and if the table size is smaller than the user-specified maximum RAM (**Max. Storage**), a new table entry is added.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Table entries are stored as leaves in a binary tree. When a new table entry is *added*, the original leaf becomes a node with two leaves—the original leaf and the new entry. A cutting hyper-plane is created at the new node, so that the two leaves are on either side of this cutting plane. A composition vector ϕ_q^0 will hence lie on either side of this hyper-plane.

The ISAT algorithm is summarized as follows:

1. The ISAT table is *queried* for every composition vector during the reaction step.
2. For each query ϕ_q^0 the table is traversed to identify a leaf whose composition ϕ^0 is close to ϕ_q^0.
3. If the query composition ϕ_q^0 lies within the EOA of the leaf, then the mapping ϕ_q^1 is *retrieved* using interpolation by Equation 7.169 (p. 285). Otherwise, Direct Integration (DI) is performed and the error ε between the DI and the linear interpolation is measured.
4. If the error ε is less than the tolerance, then the ellipsoid of accuracy is *grown* and the DI result is returned. Otherwise, a new table entry is *added*.

At the start of the simulation, most operations are *adds* and *grows*. Later, as more of the composition space is tabulated, *retrieves* become frequent. Since *adds* and *grows* are very slow whereas *retrieves* are relatively quick, initial Ansys Fluent iterations are slow but accelerate as the table is built.

There are two inputs to ISAT, namely the ISAT error tolerance (ε_{tol}) and the maximum ISAT table size (in Mbytes). Large values of ε_{tol} provide faster run times, but larger error. For steady-state simulations, it is advised to start the simulation with a large ε_{tol} and reduce ε_{tol} as the solution stabilizes and nears convergence. For this reason, the default ISAT error tolerance of 0.001 is relatively large and should be decreased for unsteady simulations or when a steady-state simulation converges. It is recommended that you monitor the species of interest and re-converge with reduced ε_{tol} until these species change in acceptably small increments. The maximum ISAT table size should be set to just below the available RAM memory on the computer.

7.3.3. Dynamic Mechanism Reduction

The solution time for a chemically reacting flow problem increases with the size of the reaction mechanism used. Typically, this relationship is

$$t_{CPU} \propto c_1 N_{species}^3 + c_2 N_{species}^2 + c_3 N_{species} + c_4 N_{reactions}$$

where c_1, c_2, c_3, and c_4 are constants. However, $c_4 \propto N_{species}$ if finite-difference Jacobian is applied.

Dynamic Mechanism Reduction can accelerate the simulation by decreasing the number of species ($N_{species}$) and number of reactions ($N_{reactions}$) in the chemical mechanism. In general, more reduction results in faster, but less accurate simulations. Mechanism reduction seeks to decrease the mechanism size while limiting accuracy loss to some predefined tolerances.

Unlike skeletal reduction where mechanism reduction is completed at the pre-solution stage to create a single reduced mechanism that is used throughout the simulation, Dynamic Mechanism Reduction is performed "on the fly" in every cell (or particle), at every flow iteration (for steady simulations) or time step (for transient simulations). Since the mechanism is only required to be accurate at the local

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

cell conditions, Dynamic Mechanism Reduction can be used with a higher level of reduction yet less accuracy loss than skeletal mechanism [[391] (p. 1075), [486] (p. 1081)].

Mechanism reduction in Ansys Fluent is performed using the Directed Relation Graph (DRG) approach [[389] (p. 1075), [390] (p. 1075)], described next.

7.3.3.1. Directed Relation Graph (DRG) Method for Mechanism Reduction

Given a list of species that need to be modeled accurately (called "targets"), DRG eliminates all species and reactions in the mechanism that do not contribute significantly (directly or indirectly) to predicting the evolution of the targets.

To generate the reduced mechanism, DRG implements the following steps.

1. DRG considers normalized contribution of each non-target species B to overall production of each target species A within a single cell:

$$r_{AB} \equiv \frac{\max_i\left(\left|\nu_{A,i}\omega_i\delta_{Bi}\right|\right)}{\max_i\left(\left|\nu_{A,i}\omega_i\right|\right)} \tag{7.170}$$

   where

$$\delta_{Bi} = \begin{cases} 1, & \text{If reaction } i \text{ involves species } B \\ 0, & \text{otherwise} \end{cases}$$

   $\omega_i(T,P,Y)$ is a chemical rate of elementary reaction i at a local cell condition defined by the temperature T, the pressure P, and the species mass fraction Y

   $\nu_{A,i}$ is a stoichiometric coefficient of species A in reaction i

   Species B is retained in the mechanism if and only if the largest reaction rate involving both species, A and B, is larger than a fraction ε of the largest reaction rate involving species A within the cell:

$$r_{AB} > \varepsilon \tag{7.171}$$

   where ε is a specified error tolerance.

   Normalized contribution value r_{Ak} is computed for every non-target species k in the mechanism, to identify all non-target species that directly contribute significantly to modeling target species A. These species constitute the *dependent set* of A. This process is repeated for all target species, and a comprehensive dependent set is created as the union of individual dependent sets for all targets.

2. In the next step, DRG algorithm, in a similar manner, identifies indirect contributors, which are species that directly impact the dependent set, rather than the targets. In other words, if species B is included in the comprehensive dependent set created in step 1, then all the remaining species k with $r_{Bk} > \varepsilon$ must also be included. The procedure outlined in step 1 is applied to each species B in the dependent set to generate the list of species that contribute to its production or consumption.

   This process is continued for each species added to the mechanism, until no new species qualifies to be added to the comprehensive dependent set.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

The resulting set of species (including the targets) constitutes the species retained in the mechanism. All the other species, that is all the species B for which $r_{Bi}<\varepsilon$ for all species i included in the resulting set of species, are considered unimportant and are eliminated from the mechanism.

3. Finally, all reactions that do not involve any of the retained species are also eliminated from the mechanism. The resulting mechanism is the final reduced mechanism.

   A lower-dimensional ODE system is solved, involving only the retained species and reactions. The eliminated species mass fractions are stored for computing mixture quantities such as density and heat capacity.

It is important to note that computational expense of the DRG method has been shown to be linear in the number of reactions $N_{reactions}$ in the full detailed mechanism. This small additional overhead is typically significantly outweighed by the acceleration due to Dynamic Mechanism Reduction using DRG.

Mechanism reduction is controlled by the following two parameters in Ansys Fluent:

- Error Tolerance ε

  The default value for error tolerance ε is `0.01`.

- Target Species List (adjustable only in expert mode, accessible in the TUI)

  The default list of target species consists of 3 constituents. Hydrogen radical is explicitly defined as the first target species. Hydrogen radical is used as a default target species because it is strongly linked to heat release in combustion; accurate prediction of this species should ensure accurate prediction of heat release. Two other species with the largest mass fractions are added to the target species list by DRG algorithm at each time step or flow iteration.

There is also an option to remove a species from the target list whenever its mass fraction is below a given threshold. This option is disabled by default in Ansys Fluent (that is, the default value for minimum mass fraction is 0).

In general, a smaller value of error tolerance ε and a larger value of number of targets N_{target} yield larger, more accurate mechanisms, but slower simulations. The default settings should balance accuracy and efficiency well for most simulations. However, these parameters may require more careful selection for some problems, such as auto-ignition of highly complex fuels. For further instructions on how to use Dynamic Mechanism Reduction, see Using Dynamic Mechanism Reduction in the *Fluent User's Guide*.

7.3.4. Chemistry Agglomeration

Reacting flow computations with detailed mechanisms can be computationally demanding, even with the speedup provided by ISAT. Chemistry agglomeration (CA) provides additional run-time improvement, with a corresponding additional decrease in accuracy. The idea behind chemistry agglomeration is to collect cells (or particles for the Lagrangian PDF Transport model) that are close in composition space, average these to a single composition, call the reaction step integrator, then map this reaction step back to the cells. In summary:

1. Just before the reaction step, bin CFD cells that are close in composition space.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

2. Average their compositions.

3. Call ISAT to perform the chemistry integration with the single, averaged composition.

4. Map this reaction step back to the cells in the group.

The number of calls to the relatively costly chemistry integration routine (ISAT) is less than the number of cells in the domain. Chemistry agglomeration is hence similar to Reactor Network, or Multi-Zone models, which calculate reaction on a smaller number of zones than the number of cells in the CFD simulation.

A reaction mapping is represented as,

$$\phi \rightarrow \phi^{*} \tag{7.172}$$

where ϕ and ϕ^{*} are the thermo-chemical compositions (temperature, pressure, and species fractions) before and after reaction, respectively. By default, Ansys Fluent loops over all cells (or particles for the Lagrangian PDF Transport model) and calculates the reaction mapping.

Chemistry agglomeration loops through all CFD cells before reaction and bins those that are 'close' in composition space to within a specified tolerance. The composition in each bin is averaged as,

$$\overline{\phi} = \frac{\sum_i \phi_i \rho_i V_i}{\sum_i \rho_i V_i} \tag{7.173}$$

where $\overline{\phi}$ represents the agglomerated composition, subscript i is an index over the cells in the bin, and ρ_i and V_i denote the density and volume of the i^{th} cell, respectively.

After agglomeration, the ISAT reaction mapping routine is called,

$$\overline{\phi} \rightarrow \overline{\phi}^{*} \tag{7.174}$$

Finally, the reacted cluster composition is mapped back to all the CFD cells in the corresponding bin,

$$\phi_i^{*} = \phi_i + \left(\overline{\phi}^{*} - \overline{\phi}\right) \tag{7.175}$$

for every cell i within the bin.

To ensure an iso-enthalpic reaction, the enthalpy of each cell after reaction, h_i^{*}, is set according to:

$$h_i^{*} = h_i + \sum_k h_k^{o}\left(\overline{Y}_k^{\,*} - Y_{i,k}\right) \tag{7.176}$$

where h_i denotes the enthalpy of cell i before the reaction step, h_k^{o} is the formation enthalpy of species k, $\overline{Y}_k^{\,*}$ is the mass fraction after reaction for the cell cluster, and $Y_{i,k}$ is the cell mass fraction before reaction.

To learn how to enable Chemistry Agglomeration, see Using Chemistry Agglomeration in the User's Guide.

For more information, see the following section:

7.3.4.1. Binning Algorithm

7.3.4.1. Binning Algorithm

A uniform Cartesian mesh is used to bin cells at every reaction step. The composition space has $N=N_{spe}+2$ dimensions, where N_{spe} are the number of species and the additional two dimensions are temperature and pressure. To reduce the cost of tabulating in N-dimensions, a subset of M representative composition space coordinates is selected.

For tabulation purposes, each coordinate k of the M-dimensional reduced composition space is normalized as,

$$\hat{\phi}_k=\frac{\phi_k-\phi_k^{min}}{\phi_k^{max}-\phi_k^{min}} \tag{7.177}$$

where ϕ_k^{min} and ϕ_k^{max} are the minimum and maximum of composition space variable k in the CFD computational domain. Hence, the reduced composition space binning mesh extends from zero to one in each of the M coordinates. The hyper-cube is discretized into $L_{sp}=1/\varepsilon_{tol}$ uniform intervals along each of the species or pressure coordinates and $L_T=(T_{\max}-T_{\min})/\Delta_T$ uniform temperature intervals. Here, ε_{tol} (dimensionless) and Δ_T (units of Kelvin) are user specified CA tolerances that indicate maximum bin sizes for species and temperature dimensions, respectively. All CFD cell compositions that fall within a reduced composition space bin are agglomerated.

Since most of the bins in the M-dimensional hyper-cube are likely to be empty, a dynamic hash table [30] (p. 1054) is employed to efficiently store the bins. The hash table maps a unique bin index in the M-dimensional hyper-cube to a 1D line. Since the reduced space hyper-cube is discretized into equi-spaced bins, a unique index for every bin can be defined as,

$$j=\sum_{l=0}^{M-1} i_l L^l \tag{7.178}$$

where $i_l=\hat{\phi}_l L$ is the bin index in the l^{th} dimension.

The hash table maps j to a 1D table of size H, which is much smaller than the number of entries in the reduced space hyper-cube, L^M. A simple hash mapping function is the modulo function:

$$h=j \bmod H \tag{7.179}$$

Different M-space indices j may have the same hash mapping index h, which results in collisions. In the dynamic hash table, the number of collisions, as well as the number of empty hash table entries are monitored, and the hash table size, H, is adjusted at each iteration to set these to be within acceptable limits.

The M compositions representing the reduced space coordinates are selected from the $N_{spe}+2$ dimensional composition space as follows. By default, temperature is included and pressure is excluded. The current algorithm orders the species with largest total mass in the computational domain. For example, N_2 is typically first in this list for combustors using air as the oxidizer. The first $M-1$ species of this list are selected as the reduced composition space variables.

By selecting species with the largest mass, minor species are invariably neglected. Since CO and OH are of interest in many practical applications, CO and OH are inserted at the front of the list if they exist in the computational domain. It was found that inclusion of these species in the table coordinates consistently provided increased accuracy for all thermo-chemical variables.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

In summary, the default coordinate algorithm selects T as a reduced space coordinate, then OH (if OH is present in the CFD domain), then CO (if CO is present in the CFD domain), followed by the species with largest mass in the domain until M variables are reached (with M=4 as default). The default reduced table size (M=4) and species list should be suitable for most reacting flow applications, but can be changed. Contact your technical support engineer to learn how to make the changes.

7.3.5. Chemical Mechanism Dimension Reduction

Dimension Reduction is a chemistry acceleration method in addition to ISAT storage-retrieval, Dynamic Mechanism Reduction, and Cell Agglomeration, providing faster chemistry calculations with a corresponding loss of accuracy. To learn how to use chemical mechanism Dimension Reduction, see Dimension Reduction in the User's Guide.

Dimension Reduction solves a smaller number (N_r) of species transport equations (see Equation 7.1 (p. 238)) than the number of species in the full chemical mechanism (N_f). These are termed the represented species. The remaining $N_f - N_r$ species are termed the unrepresented species. In Ansys Fluent, the unrepresented species are reconstructed assuming that they are in chemical equilibrium [524] (p. 1083). The Dimension Reduction algorithm is implemented as follows:

- Ansys Fluent solves transport equations for the mixture enthalpy and the N_r represented species mass fractions. Transport equations are also solved for the unrepresented element mass fractions, which are the cumulative mass fraction of each atomic element in all the unrepresented species.
- For the reaction step, the initial unrepresented species in a cell are reconstructed by assuming that they are in chemical equilibrium at the cell pressure and enthalpy, subject to the constraints of the represented species mass fractions and the unrepresented element mass fractions. That is, the unrepresented species are determined as those that give maximum mixture entropy with the represented species held fixed, while also satisfying the unrepresented element mass fractions and mixture enthalpy. At the end of this reconstruction step, all N_f species mass fractions of the full mechanism are available in the cell.
- Next, the full detailed reaction mechanism with all N_f species is integrated for the reaction time step. Reaction mappings for the N_r represented species are available after the reaction step, and the mappings for the unrepresented species are discarded.

As Dimension Reduction integrates the full chemical mechanism, which is computationally expensive, its advantage lies in coupling with ISAT. An ISAT table with N_r dimensions will cover more of the reaction manifold than ISAT tabulation of the full N_f dimensions in the same time. Consequently, a simulation with Dimension Reduction is initially not much faster than a simulation with the full mechanism, but iterations can be substantially faster at later times when the ISAT table is retrieved. Additionally, the retrieval time with Dimension Reduction enabled (proportional to N_r^2), is much smaller than that with the Full Mechanism (proportional to N_f^2). For this reason, Dimension Reduction is only available with ISAT.

Note that Dimension Reduction allows CFD simulations with full chemical mechanisms consisting of more than the Ansys Fluent transported species limit of 700, as long as the number of represented species is less than 700.

For more information, see the following section:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

7.3.5.1. Selecting the Represented Species

7.3.5.1. Selecting the Represented Species

Judicious selection of the represented species is important for simulation accuracy. Boundary and initial species must be included in the represented species list. Species of interest, especially species that are far from chemical equilibrium, such as pollutants, should also be included. Intermediate species that occur in large mass fractions relative to the fuel and oxidizer species should be included, as well as species important in the chemical pathway. For example, for methane combustion in air, CH_3 should be included as a represented species since CH_4 pyrolizes to CH_3 first.

In Ansys Fluent, you will specify the number of represented species, as well as selected represented species, such as the boundary **Fuel** and **Oxidizer** species. When the selected represented species are less than the input number of represented species, the remaining represented species are taken from the full mechanism in the order they appear in the **Mixture Species** list.

After you obtain a preliminary solution with **Dimension Reduction**, it is a good idea to check the magnitude of all unrepresented species, which are available in the **Full Mechanism Species...** option in the **Contours** dialog box. If the mass fraction of an unrepresented species is larger than other represented species, you should repeat the simulation with this species included in the represented species list. In turn, the mass fraction of unrepresented elements should decrease.

7.3.6. Dynamic Cell Clustering with Ansys Fluent CHEMKIN-CFD Solver

For simulations that involve direct use of finite-rate chemistry, the chemistry-solution portion of the species equations is solved on a cell-by-cell basis. As a result, the equations solved are independent of the mass and volume of a specific cell. In this way, cells that have the same temperature, pressure, and initial species mass fractions will yield the same result. To take advantage of this fact, when you select the Ansys CHEMKIN-CFD Solver, Ansys Fluent uses a Dynamic Cell Clustering (DCC) method to group computational cells of high similarity into clusters using an efficient data-clustering method. That requires solution of the kinetic equations only once for each cluster. The optimal number of clusters is dynamically determined for each CFD time step. The clustering algorithm is solely based on the cell thermochemical states and is independent of their locations in the CFD mesh. In addition, the clustering algorithm is highly automated and requires minimal inputs. The algorithm uses cell temperature and equivalence ratio as the clustering indices.

The DCC method consists of three major steps:

1. Grouping cells into clusters using an evolutionary data-clustering algorithm.

2. Solving chemical kinetic equations based on cluster averaged state variables.

3. Mapping the cluster averaged solution back to the individual cells while preserving the initial temperature and species stratification.

Further information of dynamic cell clustering can be found in section Using Dynamic Cell Clustering in the *Fluent User's Guide*.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

7.3.7. Dynamic Adaptive Chemistry with Ansys Fluent CHEMKIN-CFD Solver

To ensure validity over a wide range of thermochemical conditions, comprehensive kinetic mechanisms for the combustion of realistic fuels typically include hundreds of species and thousands of elementary reactions. However, much smaller subsets of species and reactions are often adequate to capture the dominant reaction pathways for specific local conditions over a short time span (typically taken to be the hydrodynamic time step in CFD calculations). Ansys Fluent offers the dynamic adaptive chemistry (DAC) method [364] (p. 1074), [365] (p. 1074) which makes use of this fact. The method reduces the comprehensive detailed mechanism to locally valid smaller mechanisms. This operation is performed on the fly (that is, during the dynamic simulation, at every time step). It is based on a skeletal mechanism-reduction method called Directed-Relation-Graph with Error Propagation (DRGEP) method [389] (p. 1075), which offers very efficient and very accurate reduction. The Dynamic Adaptive Chemistry is similar to the Dynamic Mechanism Reduction (see Dynamic Mechanism Reduction (p. 286)), but only available with the CHEMKIN-CFD Solver. For details about the theory behind this method, see [364] (p. 1074), [365] (p. 1074), and [389] (p. 1075).

For information about using the Dynamic Adaptive Chemistry method, see Using Dynamic Adaptive Chemistry with Ansys Fluent CHEMKIN-CFD Solver in the *Fluent User's Guide*.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information
of ANSYS, Inc. and its subsidiaries and affiliates.

Chapter 8: Modelling of Turbulent Combustion With Reduced Order

This chapter contains the theory of turbulent combustion with reduced order used in Ansys Fluent:

8.1. Non-Premixed Combustion

8.2. Premixed Combustion

8.3. Partially Premixed Combustion

8.1. Non-Premixed Combustion

In non-premixed combustion, fuel and oxidizer enter the reaction zone in distinct streams. This is in contrast to premixed systems, in which reactants are mixed at the molecular level before burning. Examples of non-premixed combustion include pulverized coal furnaces, diesel internal-combustion engines and pool fires.

Under certain assumptions, the thermochemistry can be reduced to a single parameter: the mixture fraction. The mixture fraction, denoted by f, is the mass fraction that originated from the fuel stream. In other words, it is the local mass fraction of burnt and unburnt fuel stream elements (C, H, and so on) in all the species (CO_2, H_2O, O_2, and so on). The approach is elegant because atomic elements are conserved in chemical reactions. In turn, the mixture fraction is a conserved scalar quantity, and therefore its governing transport equation does not have a source term. Combustion is simplified to a mixing problem, and the difficulties associated with closing nonlinear mean reaction rates are avoided. Once mixed, the chemistry can be modeled as being in chemical equilibrium with the Equilibrium model, being near chemical equilibrium with the Steady Diffusion Flamelet model, or significantly departing from chemical equilibrium with the Unsteady Diffusion Flamelet model.

For more information about using the non-premixed combustion model, see Modeling Non-Premixed Combustion in the User's Guide. Theoretical information about the non-premixed combustion model is presented in the following sections:

8.1.1. Introduction

8.1.2. Non-Premixed Combustion and Mixture Fraction Theory

8.1.3. Restrictions and Special Cases for Using the Non-Premixed Model

8.1.4. The Diffusion Flamelet Models Theory

8.1.5. The Steady Diffusion Flamelet Model Theory

8.1.6. The Unsteady Diffusion Flamelet Model Theory

8.1.1. Introduction

Non-premixed modeling involves the solution of transport equations for one or two conserved scalars (the mixture fractions). Equations for individual species are not solved. Instead, species concentrations

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

are derived from the predicted mixture fraction fields. The thermochemistry calculations are preprocessed and then tabulated for look-up in Ansys Fluent. Interaction of turbulence and chemistry is accounted for with an assumed-shape Probability Density Function (PDF).

8.1.2. Non-Premixed Combustion and Mixture Fraction Theory

Information about non-premixed combustion and mixture fraction theory are presented in the following sections:

8.1.2.1. Mixture Fraction Theory

8.1.2.2. Modeling of Turbulence-Chemistry Interaction

8.1.2.3. Non-Adiabatic Extensions of the Non-Premixed Model

8.1.2.4. Chemistry Tabulation

8.1.2.1. Mixture Fraction Theory

8.1.2.1.1. Definition of the Mixture Fraction

The basis of the non-premixed modeling approach is that under a certain set of simplifying assumptions, the instantaneous thermochemical state of the fluid is related to a conserved scalar quantity known as the mixture fraction, f. The mixture fraction can be written in terms of the atomic mass fraction as [598] (p. 1087):

$$f = \frac{Z_i - Z_{i,ox}}{Z_{i,fuel} - Z_{i,ox}} \tag{8.1}$$

where Z_i is the elemental mass fraction for element, i. The subscript *ox* denotes the value at the oxidizer stream inlet and the subscript fuel denotes the value at the fuel stream inlet. If the diffusion coefficients for all species are equal, then Equation 8.1 (p. 296) is identical for all elements, and the mixture fraction definition is unique. The mixture fraction is therefore the elemental mass fraction that originated from the fuel stream.

If a secondary stream (another fuel or oxidant, or a non-reacting stream) is included, the fuel and secondary mixture fractions are simply the elemental mass fractions of the fuel and secondary streams, respectively. The sum of all three mixture fractions in the system (fuel, secondary stream, and oxidizer) is always equal to 1:

$$f_{fuel} + f_{sec} + f_{ox} = 1 \tag{8.2}$$

This indicates that only points on the plane ABC (shown in Figure 8.1: Relationship of Mixture Fractions (Fuel, Secondary Stream, and Oxidizer) (p. 297)) in the mixture fraction space are valid. Consequently, the two mixture fractions, f_{fuel} and f_{sec}, cannot vary independently; their values are valid only if they are both within the triangle OBC shown in Figure 8.1: Relationship of Mixture Fractions (Fuel, Secondary Stream, and Oxidizer) (p. 297).

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Figure 8.1: Relationship of Mixture Fractions (Fuel, Secondary Stream, and Oxidizer)

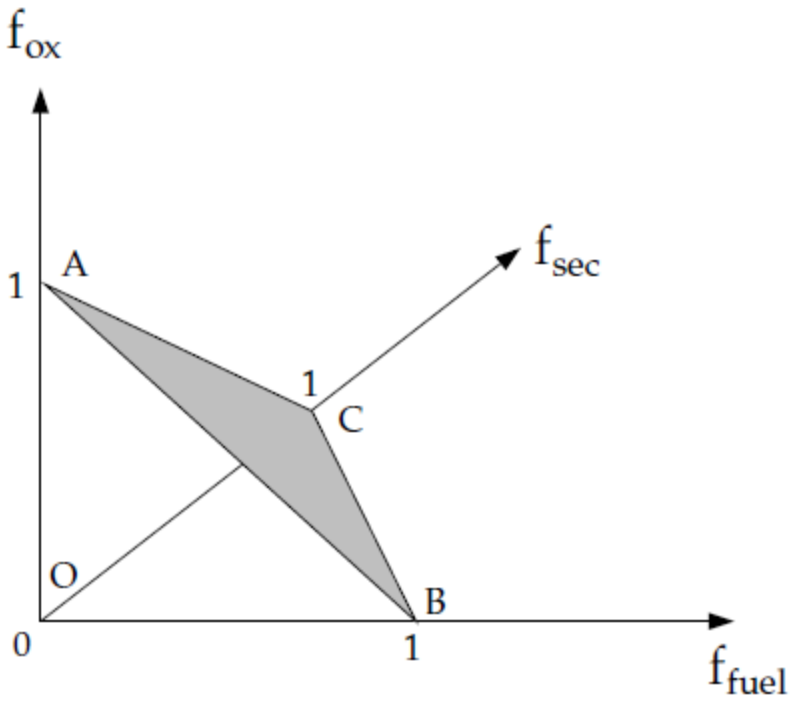
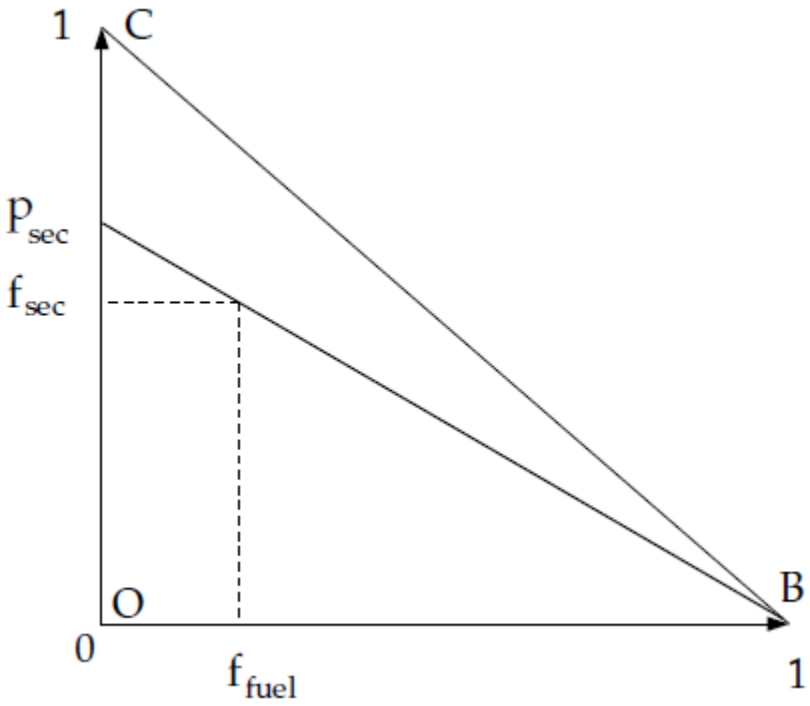


Figure 8.2: Relationship of Mixture Fractions (Fuel, Secondary Stream, and Normalized Secondary Mixture Fraction)



Ansys Fluent discretizes the triangle OBC as shown in Figure 8.2: Relationship of Mixture Fractions (Fuel, Secondary Stream, and Normalized Secondary Mixture Fraction) (p. 297). Essentially, the primary mixture fraction, f_{fuel}, is allowed to vary between zero and one, as for the single mixture fraction case, while the secondary mixture fraction lies on lines with the following equation:

$$f_{sec}=p_{sec}\times\left(1-f_{fuel}\right) \tag{8.3}$$

where p_{sec} is the normalized secondary mixture fraction and is the value at the intersection of a line with the secondary mixture fraction axis. Note that unlike f_{sec}, p_{sec} is bounded between zero and one, regardless of the f_{fuel} value.

An important characteristic of the normalized secondary mixture fraction, p_{sec}, is its assumed statistical independence from the fuel mixture fraction, f_{fuel}. Note that unlike f_{sec}, p_{sec} is not a conserved scalar. This normalized mixture fraction definition, p_{sec}, is used everywhere in Ansys Fluent when prompted for Secondary Mixture Fraction except when defining the rich limit for a secondary fuel stream, which is defined in terms of f_{sec}.

8.1.2.1.2. Transport Equations for the Mixture Fraction

Under the assumption of equal diffusivities, the species equations can be reduced to a single equation for the mixture fraction, f. The reaction source terms in the species equations cancel (since elements are conserved in chemical reactions), and therefore f is a conserved quantity. While the assumption of equal diffusivities is problematic for laminar flows, it is generally acceptable for turbulent flows where turbulent convection overwhelms molecular diffusion. The Favre mean (density-averaged) mixture fraction equation is

$$\frac{\partial}{\partial t}\left(\rho\overline{f}\right)+\nabla\cdot\left(\rho\vec{v}\overline{f}\right)=\nabla\cdot\left(\left(\frac{k}{C_p}+\frac{\mu_t}{\sigma_t}\right)\nabla\overline{f}\right)+S_m+S_{user} \tag{8.4}$$

where k is laminar thermal conductivity of the mixture, C_p is the mixture specific heat, σ_t is the Prandtl number, and μ_t is the turbulent viscosity.

The source term S_m is due solely to transfer of mass into the gas phase from liquid fuel droplets or reacting particles (for example, coal). S_{user} is any user-defined source term.

In addition to solving for the Favre mean mixture fraction, Ansys Fluent solves a conservation equation for the mixture fraction variance, $\overline{f'^2}$ [282] (p. 1069):

$$\frac{\partial}{\partial t}\left(\rho\overline{f'^2}\right)+\nabla\cdot\left(\rho\vec{v}\overline{f'^2}\right)=\nabla\cdot\left(\left(\frac{k}{C_p}+\frac{\mu_t}{\sigma_t}\right)\nabla\overline{f'^2}\right)+C_g\mu_t\cdot\left(\nabla\overline{f}\right)^2-C_d\rho\frac{\varepsilon}{k}\overline{f'^2}+S_{user} \tag{8.5}$$

where k is laminar thermal conductivity of the mixture, C_p is the mixture specific heat, σ_t is the Prandtl number, and $f'=f-\overline{f}$. The default values for the constants σ_t, C_g, and C_d are 0.85, 2.86, and 2.0, respectively, and S_{user} is any user-defined source term.

The mixture fraction variance is used in the closure model describing turbulence-chemistry interactions (see Modeling of Turbulence-Chemistry Interaction (p. 301)).

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

For a two-mixture-fraction problem, $\overline{f_{fuel}}$ and $\overline{f'^2_{fuel}}$ are obtained from Equation 8.4 (p. 298) and Equation 8.5 (p. 298) by substituting $\overline{f_{fuel}}$ for $\overline{f}$ and $\overline{f'^2_{fuel}}$ for $\overline{f'^2}$. $\overline{f_{sec}}$ is obtained from Equation 8.4 (p. 298) by substituting $\overline{f_{sec}}$ for $\overline{f}$. $\overline{p_{sec}}$ is then calculated using Equation 8.3 (p. 298), and $\overline{p'^2_{sec}}$ is obtained by solving Equation 8.5 (p. 298) with $\overline{p_{sec}}$ substituted for $\overline{f}$. To a first-order approximation, the variances in $\overline{p_{sec}}$ and $\overline{f_{sec}}$ are relatively insensitive to $\overline{f_{fuel}}$, and therefore the equation for $\overline{p'^2_{sec}}$ is essentially the same as $\overline{f'^2_{sec}}$.

Important:

The equation for $\overline{p'^2_{sec}}$ instead of $\overline{f'^2_{sec}}$ is valid when the mass flow rate of the secondary stream is relatively small compared with the total mass flow rate.

8.1.2.1.3. The Non-Premixed Model for LES

For Large Eddy Simulations, the transport equation is not solved for the mixture fraction variance. Instead, it is modeled as

$$\overline{f'^2} = C_{var} L_s^2 \left| \nabla \overline{f} \right|^2 \tag{8.6}$$

where

C_{var} = constant

L_s = subgrid length scale (see Equation 4.307 (p. 121))

The constant C_{var} is computed dynamically when the **Dynamic Stress** option is enabled in the **Viscous** dialog box, else a constant value (with a default of 0.5) is used.

If the **Dynamic Scalar Flux** option is enabled, the turbulent Sc (σ_t in Equation 8.4 (p. 298)) is computed dynamically.

8.1.2.1.4. The Non-Premixed Model with the SBES Turbulence Model

With the SBES turbulence model, clearly identifiable RANS and LES regions are present in the domain. When the transition from RANS to LES and vice versa occurs, the mixture fraction variance from the RANS formulation calculated using Equation 8.5 (p. 298) and the mixture fraction variance from the LES formulation modeled using Equation 8.6 (p. 299) will not recover the identical behavior. To ensure consistent transition between RANS and LES models, the transport equation of the mixture fraction variance is used in the RANS region, while the algebraic formulation is used in the LES region. The blending of the two is done in the same way as for turbulent viscosity (Equation 4.297 (p. 117)). The mixture fraction variance for SBES is modeled as:

$$\overline{f'^2} = \overline{f'^2_{RANS}} f_{SDES} + \left(1 - f_{SDES}\right) \overline{f'^2_{LES}} \tag{8.7}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

8.1.2.1.5. Mixture Fraction vs. Equivalence Ratio

The mixture fraction definition can be understood in relation to common measures of reacting systems. Consider a simple combustion system involving a fuel stream (F), an oxidant stream (O), and a product stream (P) symbolically represented at stoichiometric conditions as

$$F+rO\rightarrow\left(1+r\right)P \tag{8.8}$$

where r is the air-to-fuel ratio on a mass basis. Denoting the equivalence ratio as ϕ, where

$$\phi=\frac{\left(fuel/\ air\right)_{actual}}{\left(fuel/\ air\right)_{stoichiometric}} \tag{8.9}$$

the reaction in Equation 8.8 (p. 300), under more general mixture conditions, can then be written as

$$\phi F+rO\rightarrow\left(\phi+r\right)P \tag{8.10}$$

Looking at the left side of this equation, the mixture fraction for the system as a whole can then be deduced to be

$$f=\frac{\phi}{\phi+r} \tag{8.11}$$

Equation 8.11 (p. 300) allows the computation of the mixture fraction at stoichiometric conditions (ϕ = 1) or at fuel-rich conditions (for example, ϕ > 1), or fuel-lean conditions (for example, ϕ < 1).

8.1.2.1.6. Relationship of Mixture Fraction to Species Mass Fraction, Density, and Temperature

The power of the mixture fraction modeling approach is that the chemistry is reduced to one or two conserved mixture fractions. Under the assumption of chemical equilibrium, all thermochemical scalars (species fractions, density, and temperature) are uniquely related to the mixture fraction(s).

For a single mixture fraction in an adiabatic system, the instantaneous values of mass fractions, density, and temperature depend solely on the instantaneous mixture fraction, f:

$$\phi_i=\phi_i\left(f\right) \tag{8.12}$$

If a secondary stream is included, the instantaneous values will depend on the instantaneous fuel mixture fraction, f_{fuel}, and the secondary partial fraction, p_{sec}:

$$\phi_i=\phi_i\left(f_{fuel},p_{sec}\right) \tag{8.13}$$

In Equation 8.12 (p. 300) and Equation 8.13 (p. 300), ϕ_i represents the instantaneous species mass fraction, density, or temperature. In the case of non-adiabatic systems, the effect of heat loss/gain is parameterized as

$$\phi_i=\phi_i\left(f,H\right) \tag{8.14}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

for a single mixture fraction system, where H is the instantaneous enthalpy (see Equation 5.8 (p. 173)).

If a secondary stream is included,

$$\phi_i = \phi_i\left(f_{fuel}, p_{sec}, H\right) \tag{8.15}$$

Examples of non-adiabatic flows include systems with radiation, heat transfer through walls, heat transfer to/from discrete phase particles or droplets, and multiple inlets at different temperatures. Additional detail about the mixture fraction approach in such non-adiabatic systems is provided in Non-Adiabatic Extensions of the Non-Premixed Model (p. 305).

In many reacting systems, the combustion is not in chemical equilibrium. Ansys Fluent offers several approaches to model chemical non-equilibrium, including the finite-rate (see The Generalized Finite-Rate Formulation for Reaction Modeling (p. 243)), EDC (see The Eddy-Dissipation-Concept (EDC) Model (p. 249)), and PDF transport (see Composition PDF Transport (p. 276)) models, where detailed kinetic mechanisms can be incorporated.

There are three approaches in the non-premixed combustion model to simulate chemical non-equilibrium. The first is to use the Rich Flammability Limit (RFL) option in the Equilibrium model, where rich regions are modeled as a mixed-but-unburnt mixture of pure fuel and a leaner equilibrium burnt mixture (see Enabling the Rich Flammability Limit (RFL) Option in the User's Guide). The second approach is the Steady Diffusion Flamelet model, where chemical non-equilibrium due to diffusion flame stretching by turbulence can be modeled. The third approach is the Unsteady Diffusion Flamelet model where slow-forming product species that are far from chemical equilibrium can be modeled. See The Diffusion Flamelet Models Theory (p. 319) and The Unsteady Diffusion Flamelet Model Theory (p. 327) for details about the Steady and Unsteady Diffusion Flamelet models in Ansys Fluent.

8.1.2.2. Modeling of Turbulence-Chemistry Interaction

Equation 8.12 (p. 300) through Equation 8.15 (p. 301) describe the instantaneous relationships between mixture fraction and species fractions, density, and temperature under the assumption of chemical equilibrium. The Ansys Fluent prediction of the turbulent reacting flow, however, is concerned with prediction of the averaged values of these fluctuating scalars. How these averaged values are related to the instantaneous values depends on the turbulence-chemistry interaction model. Ansys Fluent applies the assumed-shape probability density function (PDF) approach as its closure model when the non-premixed model is used. The assumed shape PDF closure model is described in this section.

8.1.2.2.1. Description of the Probability Density Function

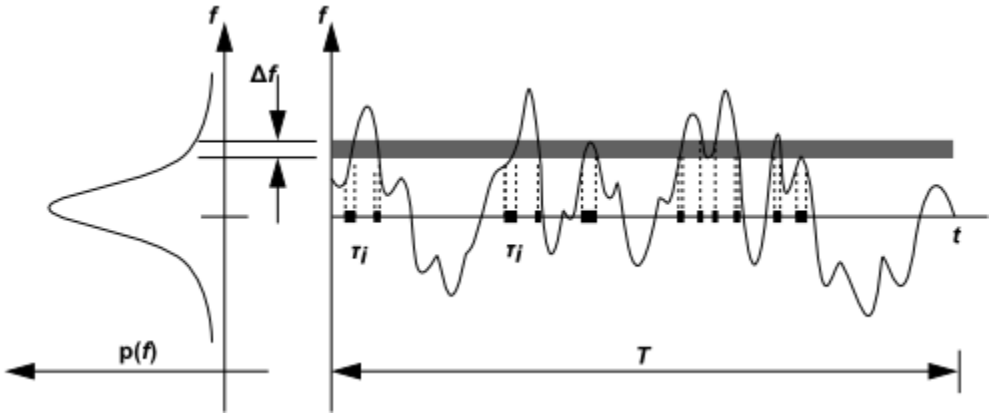
The Probability Density Function, written as $p(f)$, can be thought of as the fraction of time that the fluid spends in the vicinity of the state f. Figure 8.3: Graphical Description of the Probability Density Function (p. 302) plots the time trace of mixture fraction at a point in the flow (right-hand side) and the probability density function of f (left-hand side). The fluctuating value of f, plotted on the right side of the figure, spends some fraction of time in the range denoted as Δf. $p(f)$, plotted on the left side of the figure, takes on values such that the area under its curve in the band denoted, Δf, is equal to the fraction of time that f spends in this range. Written mathematically,

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$p(f)\Delta f = \lim_{T\to\infty}\frac{1}{T}\sum_i \tau_i \tag{8.16}$$

where T is the time scale and τ_i is the amount of time that f spends in the Δf band. The shape of the function $p(f)$ depends on the nature of the turbulent fluctuations in f. In practice, $p(f)$ is unknown and is modeled as a mathematical function that approximates the actual PDF shapes that have been observed experimentally.

Figure 8.3: Graphical Description of the Probability Density Function



8.1.2.2.2. Derivation of Mean Scalar Values from the Instantaneous Mixture Fraction

The probability density function $p(f)$, describing the temporal fluctuations of f in the turbulent flow, can be used to compute averaged values of variables that depend on f. Density-weighted mean species mass fractions and temperature can be computed (in adiabatic systems) as

$$\overline{\phi}_i = \int_0^1 p(f)\phi_i(f)df \tag{8.17}$$

for a single-mixture-fraction system. When a secondary stream exists, mean values are calculated as

$$\overline{\phi}_i = \int_0^1\int_0^1 p_1\left(f_{fuel}\right)p_2\left(p_{sec}\right)\phi_i\left(f_{fuel},p_{sec}\right)df_{fuel}dp_{sec} \tag{8.18}$$

where p_1 is the PDF of f_{fuel} and p_2 is the PDF of p_{sec}. Here, statistical independence of f_{fuel} and p_{sec} is assumed, so that $p\left(f_{fuel},p_{sec}\right)=p_1\left(f_{fuel}\right)p_2\left(p_{sec}\right)$.

Similarly, the mean time-averaged fluid density, $\overline{\rho}$, can be computed as

$$\frac{1}{\overline{\rho}} = \int_0^1 \frac{p(f)}{\rho(f)}df \tag{8.19}$$

for a single-mixture-fraction system, and

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$\frac{1}{\overline{\rho}} = \int_0^1 \int_0^1 \frac{p_1\left(f_{fuel}\right) p_2\left(p_{sec}\right)}{\rho\left(f_{fuel}, p_{sec}\right)} df_{fuel} dp_{sec} \tag{8.20}$$

when a secondary stream exists. $\rho(f)$ or $\rho\left(f_{fuel}, p_{sec}\right)$ is the instantaneous density obtained using the instantaneous species mass fractions and temperature in the ideal gas law equation.

Using Equation 8.17 (p. 302) and Equation 8.19 (p. 302) (or Equation 8.18 (p. 302) and Equation 8.20 (p. 303)), it remains only to specify the shape of the function $p(f)$ (or $p_1\left(f_{fuel}\right)$ and $p_2\left(p_{sec}\right)$) in order to determine the local mean fluid state at all points in the flow field.

8.1.2.2.3. The Assumed-Shape PDF

The shape of the assumed PDF, $p(f)$, is described in Ansys Fluent by one of two mathematical functions:

- the double-delta function (two-mixture-fraction cases only)
- the β-function (single- and two-mixture-fraction cases)

The double-delta function is the most easily computed, while the β-function most closely represents experimentally observed PDFs. The shape produced by this function depends solely on the mean mixture fraction, $\overline{f}$, and its variance, $\overline{f'^2}$. A detailed description of each function follows.

8.1.2.2.3.1. The Double Delta Function PDF

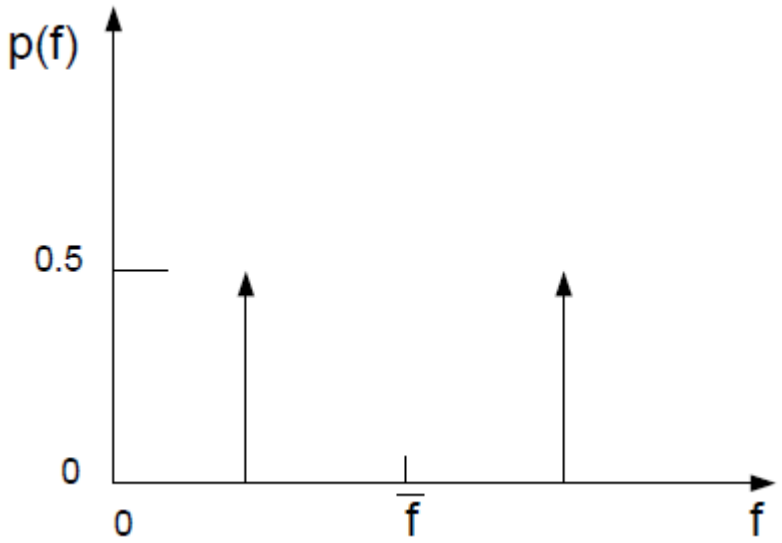
The double delta function is given by

$$p(f) = \begin{cases} 0.5, & f = \overline{f} - \sqrt{\overline{f'^2}} \\ 0.5, & f = \overline{f} + \sqrt{\overline{f'^2}} \\ 0, & \text{elsewhere} \end{cases} \tag{8.21}$$

with suitable bounding near $f = 1$ and $f = 0$. One example of the double delta function is illustrated in Figure 8.4: Example of the Double Delta Function PDF Shape (p. 304). As noted above, the double delta function PDF is very easy to compute but is invariably less accurate than the alternate β-function PDF because it assumes that only two states occur in the turbulent flow. For this reason, it is available only for two-mixture-fraction simulations where the savings in computational cost is significant.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Figure 8.4: Example of the Double Delta Function PDF Shape



8.1.2.2.3.2. The β-Function PDF

The β-function PDF shape is given by the following function of $\overline{f}$ and $\overline{f'^2}$:

$$p(f)=\frac{f^{\alpha-1}(1-f)^{\beta-1}}{\int f^{\alpha-1}(1-f)^{\beta-1}df} \tag{8.22}$$

where

$$\alpha=\overline{f}\left[\frac{\overline{f}(1-\overline{f})}{\overline{f'^2}}-1\right] \tag{8.23}$$

and

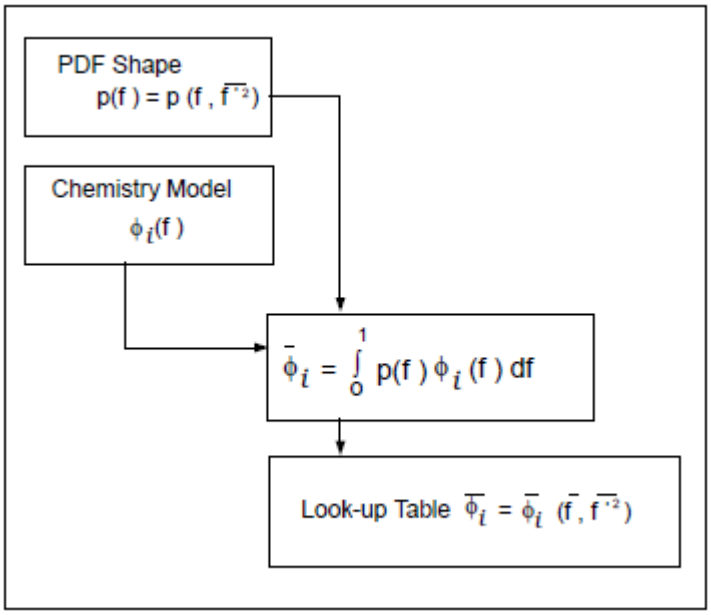
$$\beta=(1-\overline{f})\left[\frac{\overline{f}(1-\overline{f})}{\overline{f'^2}}-1\right] \tag{8.24}$$

Importantly, the PDF shape $p(f)$ is a function of only its first two moments, namely the mean mixture fraction, $\overline{f}$, and the mixture fraction variance, $\overline{f'^2}$. Thus, given Ansys Fluent's prediction of $\overline{f}$ and $\overline{f'^2}$ at each point in the flow field (Equation 8.4 (p. 298) and Equation 8.5 (p. 298)), the assumed PDF shape can be computed and used as the weighting function to determine the mean values of species mass fractions, density, and temperature using, Equation 8.17 (p. 302) and Equation 8.19 (p. 302) (or, for a system with a secondary stream, Equation 8.18 (p. 302) and Equation 8.20 (p. 303)).

This logical dependence is depicted visually in Figure 8.5: Logical Dependence of Averaged Scalars on Mean Mixture Fraction, the Mixture Fraction Variance, and the Chemistry Model (Adiabatic, Single-Mixture-Fraction Systems) (p. 305) for a single mixture fraction.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Figure 8.5: Logical Dependence of Averaged Scalars on Mean Mixture Fraction, the Mixture Fraction Variance, and the Chemistry Model (Adiabatic, Single-Mixture-Fraction Systems)



8.1.2.3. Non-Adiabatic Extensions of the Non-Premixed Model

Many reacting systems involve heat transfer through wall boundaries, droplets, and/or particles. In such flows the local thermochemical state is no longer related only to f, but also to the enthalpy, H. The system enthalpy impacts the chemical equilibrium calculation and the temperature and species of the reacting flow. Consequently, changes in enthalpy due to heat loss must be considered when computing scalars from the mixture fraction, as in Equation 8.14 (p. 300).

In such non-adiabatic systems, turbulent fluctuations should be accounted for by means of a joint PDF, $p(f,H)$. The computation of $p(f,H)$, however, is not practical for most engineering applications. The problem can be simplified significantly by assuming that the enthalpy fluctuations are independent of the enthalpy level (that is, heat losses do not significantly impact the turbulent enthalpy fluctuations). With this assumption, $p(f,H)=p(f)\delta(H-\overline{H})$ and mean scalars are calculated as

$$\overline{\phi}_i=\int_0^1 \phi_i(f,\overline{H})p(f)df \tag{8.25}$$

Determination of $\overline{\phi}_i$ in the non-adiabatic system therefore requires solution of the modeled transport equation for mean enthalpy:

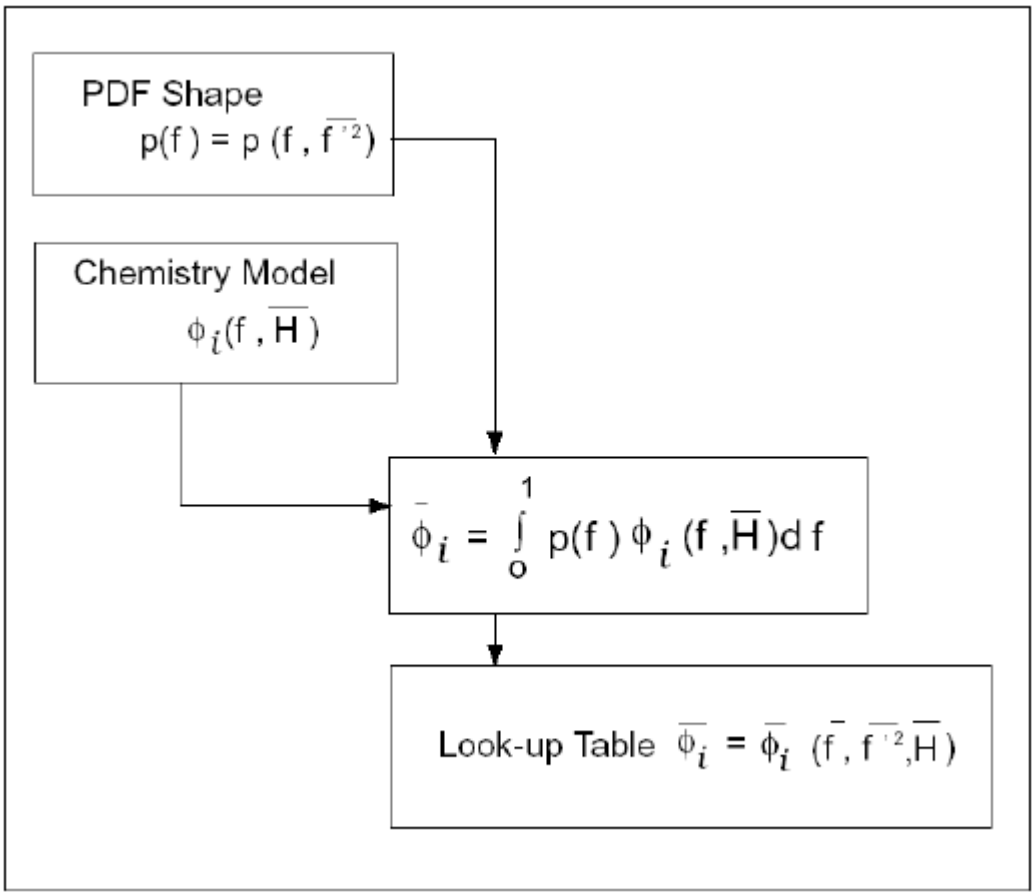
$$\frac{\partial}{\partial t}(\rho\overline{H})+\nabla\cdot(\rho\vec{v}\overline{H})=\nabla\cdot\left(\frac{k_t}{c_p}\nabla\overline{H}\right)+S_h \tag{8.26}$$

where S_h accounts for source terms due to radiation, heat transfer to wall boundaries, and heat exchange with the dispersed phase.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Figure 8.6: Logical Dependence of Averaged Scalars on Mean Mixture Fraction, the Mixture Fraction Variance, Mean Enthalpy, and the Chemistry Model (Non-Adiabatic, Single-Mixture-Fraction Systems) (p. 306) depicts the logical dependence of mean scalar values (species mass fraction, density, and temperature) on Ansys Fluent's prediction of $\overline{f}$, $\overline{f'^2}$, and $\overline{H}$ in non-adiabatic single-mixture-fraction systems.

Figure 8.6: Logical Dependence of Averaged Scalars on Mean Mixture Fraction, the Mixture Fraction Variance, Mean Enthalpy, and the Chemistry Model (Non-Adiabatic, Single-Mixture-Fraction Systems)



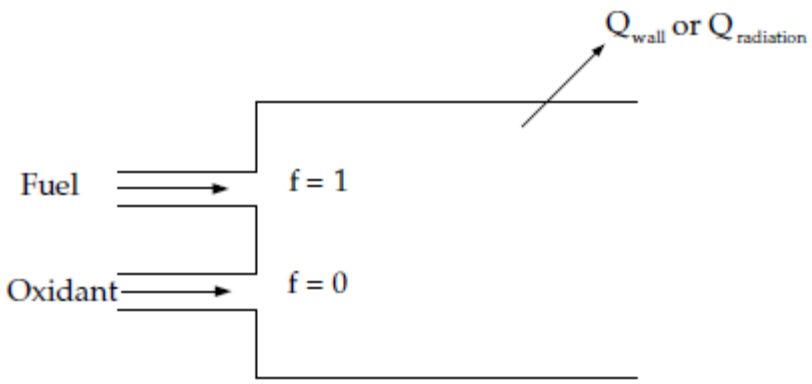
When a secondary stream is included, the mean values are calculated from

$$\overline{\phi}_i = \int_0^1 \int_0^1 \phi_i\left(f_{fuel}, p_{sec}, \overline{H}\right) p_1\left(f_{fuel}\right) p_2\left(p_{sec}\right) df_{fuel} dp_{sec} \tag{8.27}$$

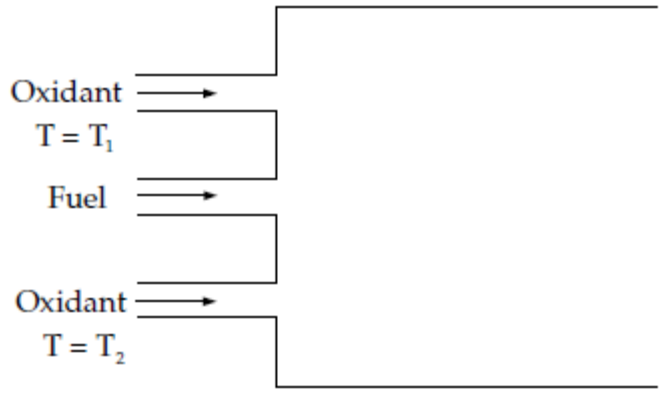
As noted above, the non-adiabatic extensions to the PDF model are required in systems involving heat transfer to walls and in systems with radiation included. In addition, the non-adiabatic model is required in systems that include multiple fuel or oxidizer inlets with different inlet temperatures. Finally, the non-adiabatic model is required in particle-laden flows (for example, liquid fuel systems or coal combustion systems) when such flows include heat transfer to the dispersed phase. Figure 8.7: Reacting Systems Requiring Non-Adiabatic Non-Premixed Model Approach (p. 307) illustrates several systems that must include the non-adiabatic form of the PDF model.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

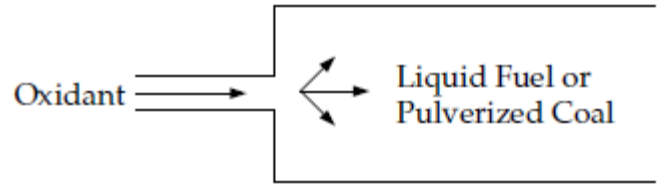
Figure 8.7: Reacting Systems Requiring Non-Adiabatic Non-Premixed Model Approach



(a) Heat Transfer to Domain Boundaries and/or Radiation Heat Transfer



(b) Multiple Fuel or Oxidant Inlets at Different Temperatures



(c) Dispersed Phase Heat or Mass Transfer (e.g., Liquid Fuel or Coal Combustion)

8.1.2.4. Chemistry Tabulation

8.1.2.4.1. Look-Up Tables for Adiabatic Systems

For an equilibrium, adiabatic, single-mixture-fraction case, the mean temperature, density, and species fraction are functions of the $\overline{f}$ and $\overline{f'^2}$ only (see Equation 8.17 (p. 302) and Equation 8.22 (p. 304)). Significant computational time can be saved by computing these integrals once, storing them in a look-up table, and retrieving them during the Ansys Fluent simulation.

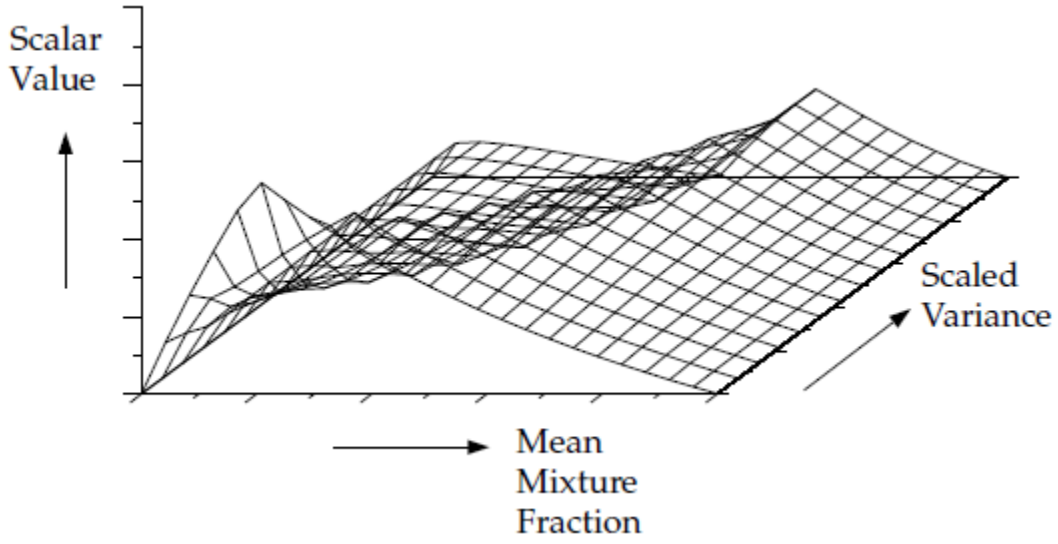
Figure 8.8: Visual Representation of a Look-Up Table for the Scalar (Mean Value of Mass Fractions, Density, or Temperature) as a Function of Mean Mixture Fraction and Mixture Fraction Variance

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

in Adiabatic Single-Mixture-Fraction Systems (p. 308) illustrates the concept of the look-up tables generated for a single-mixture-fraction system. Given Ansys Fluent's predicted value for $\overline{f}$ and $\overline{f'^2}$ at a point in the flow domain, the mean value of mass fractions, density, or temperature ($\overline{\phi}_i$) at that point can be obtained by table interpolation.

The table, Figure 8.8: Visual Representation of a Look-Up Table for the Scalar (Mean Value of Mass Fractions, Density, or Temperature) as a Function of Mean Mixture Fraction and Mixture Fraction Variance in Adiabatic Single-Mixture-Fraction Systems (p. 308), is the mathematical result of the integration of Equation 8.17 (p. 302). There is one look-up table of this type for each scalar of interest (species mass fractions, density, and temperature). In adiabatic systems, where the instantaneous enthalpy is a function of only the instantaneous mixture fraction, a two-dimensional look-up table, like that in Figure 8.8: Visual Representation of a Look-Up Table for the Scalar (Mean Value of Mass Fractions, Density, or Temperature) as a Function of Mean Mixture Fraction and Mixture Fraction Variance in Adiabatic Single-Mixture-Fraction Systems (p. 308), is all that is required.

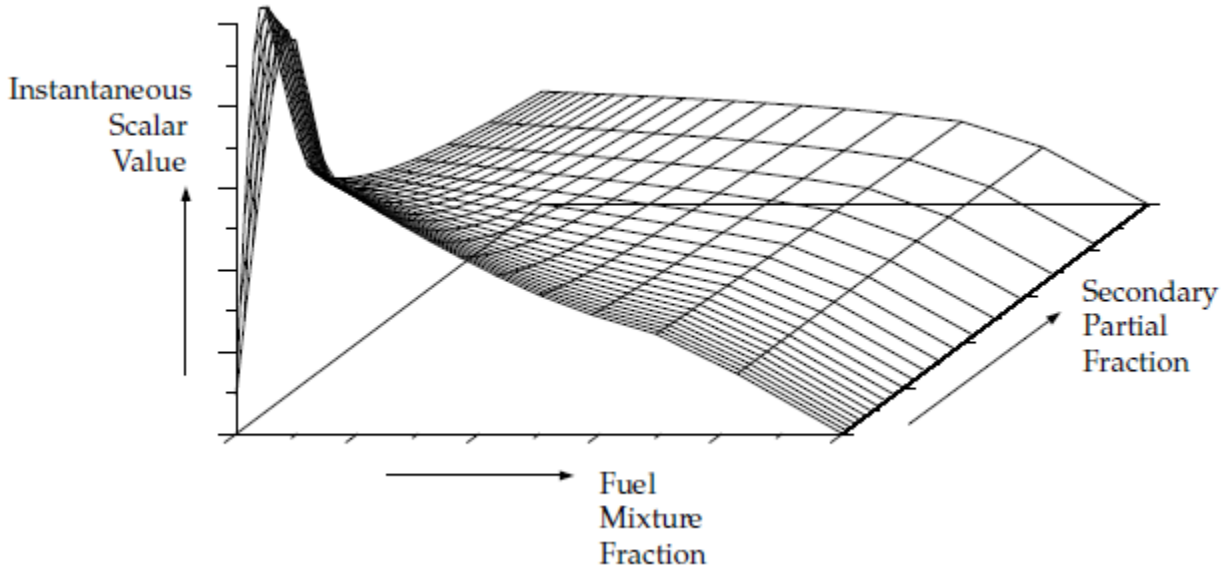
Figure 8.8: Visual Representation of a Look-Up Table for the Scalar (Mean Value of Mass Fractions, Density, or Temperature) as a Function of Mean Mixture Fraction and Mixture Fraction Variance in Adiabatic Single-Mixture-Fraction Systems



For systems with two mixture fractions, the creation and interpolation costs of four-dimensional look-up tables are computationally expensive. By default, the instantaneous properties ϕ_i are tabulated as a function of the fuel mixture fraction f_{fuel} and the secondary partial fraction p_{sec} (see Equation 8.13 (p. 300)), and the PDF integrations (see Equation 8.15 (p. 301)) are performed at run time. This two-dimensional table is illustrated in Figure 8.9: Visual Representation of a Look-Up Table for the Scalar ϕ_I as a Function of Fuel Mixture Fraction and Secondary Partial Fraction in Adiabatic Two-Mixture-Fraction Systems (p. 309). Alternatively, 4D look-up tables can be created before the simulation and interpolated at run time (see Full Tabulation of the Two-Mixture-Fraction Model in the User's Guide).

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Figure 8.9: Visual Representation of a Look-Up Table for the Scalar ϕ_I as a Function of Fuel Mixture Fraction and Secondary Partial Fraction in Adiabatic Two-Mixture-Fraction Systems



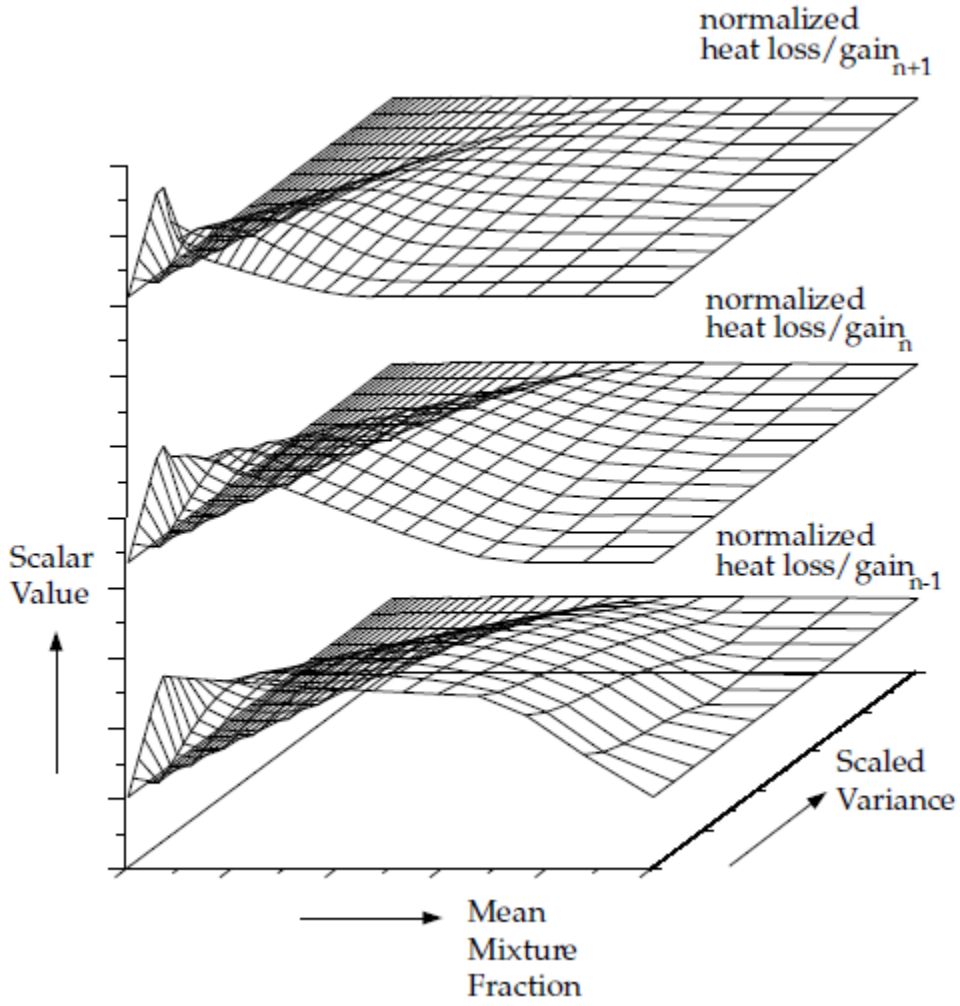
8.1.2.4.2. 3D Look-Up Tables for Non-Adiabatic Systems

In non-adiabatic systems, where the enthalpy is not linearly related to the mixture fraction, but depends also on wall heat transfer and/or radiation, a look-up table is required for each possible enthalpy value in the system. The result, for single mixture fraction systems, is a three-dimensional look-up table, as illustrated in Figure 8.10: Visual Representation of a Look-Up Table for the Scalar as a Function of Mean Mixture Fraction and Mixture Fraction Variance and Normalized Heat Loss/Gain in Non-Adiabatic Single-Mixture-Fraction Systems (p. 310), which consists of layers of two-dimensional tables, each one corresponding to a normalized heat loss or gain. The first slice corresponds to the maximum heat loss from the system, the last slice corresponds to the maximum heat gain to the system, and the zero heat loss/gain slice corresponds to the adiabatic table. Slices interpolated between the adiabatic and maximum slices correspond to heat gain, and those interpolated between the adiabatic and minimum slices correspond to heat loss.

The three-dimensional look-up table allows Ansys Fluent to determine the value of each mass fraction, density, and temperature from calculated values of $\overline{f}$, $\overline{f'^2}$, and $\overline{H}$. This three-dimensional table in Figure 8.10: Visual Representation of a Look-Up Table for the Scalar as a Function of Mean Mixture Fraction and Mixture Fraction Variance and Normalized Heat Loss/Gain in Non-Adiabatic Single-Mixture-Fraction Systems (p. 310) is the visual representation of the integral in Equation 8.25 (p. 305).

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Figure 8.10: Visual Representation of a Look-Up Table for the Scalar as a Function of Mean Mixture Fraction and Mixture Fraction Variance and Normalized Heat Loss/Gain in Non-Adiabatic Single-Mixture-Fraction Systems



For non-adiabatic, two-mixture-fraction problems, it is very expensive to tabulate and retrieve Equation 8.27 (p. 306) since five-dimensional tables are required. By default, 3D lookup tables of the instantaneous state relationship given by Equation 8.15 (p. 301) are created. The 3D table in Figure 8.11: Visual Representation of a Look-Up Table for the Scalar ϕ_I as a Function of Fuel Mixture Fraction and Secondary Partial Fraction, and Normalized Heat Loss/Gain in Non-Adiabatic Two-Mixture-Fraction Systems (p. 311) is the visual representation of Equation 8.15 (p. 301). The mean density during the Ansys Fluent solution is calculated by integrating the instantaneous density over the fuel and secondary mixture fraction space (see Equation 8.27 (p. 306)). Alternatively, 5D look-up tables can be created before the simulation and interpolated at run time (see Full Tabulation of the Two-Mixture-Fraction Model in the User's Guide). The one-time pre-generation of the 5D look-up table is very expensive, but, once built, interpolating the table during an Ansys Fluent solution is usually significantly faster than performing the integrations at run time. This is

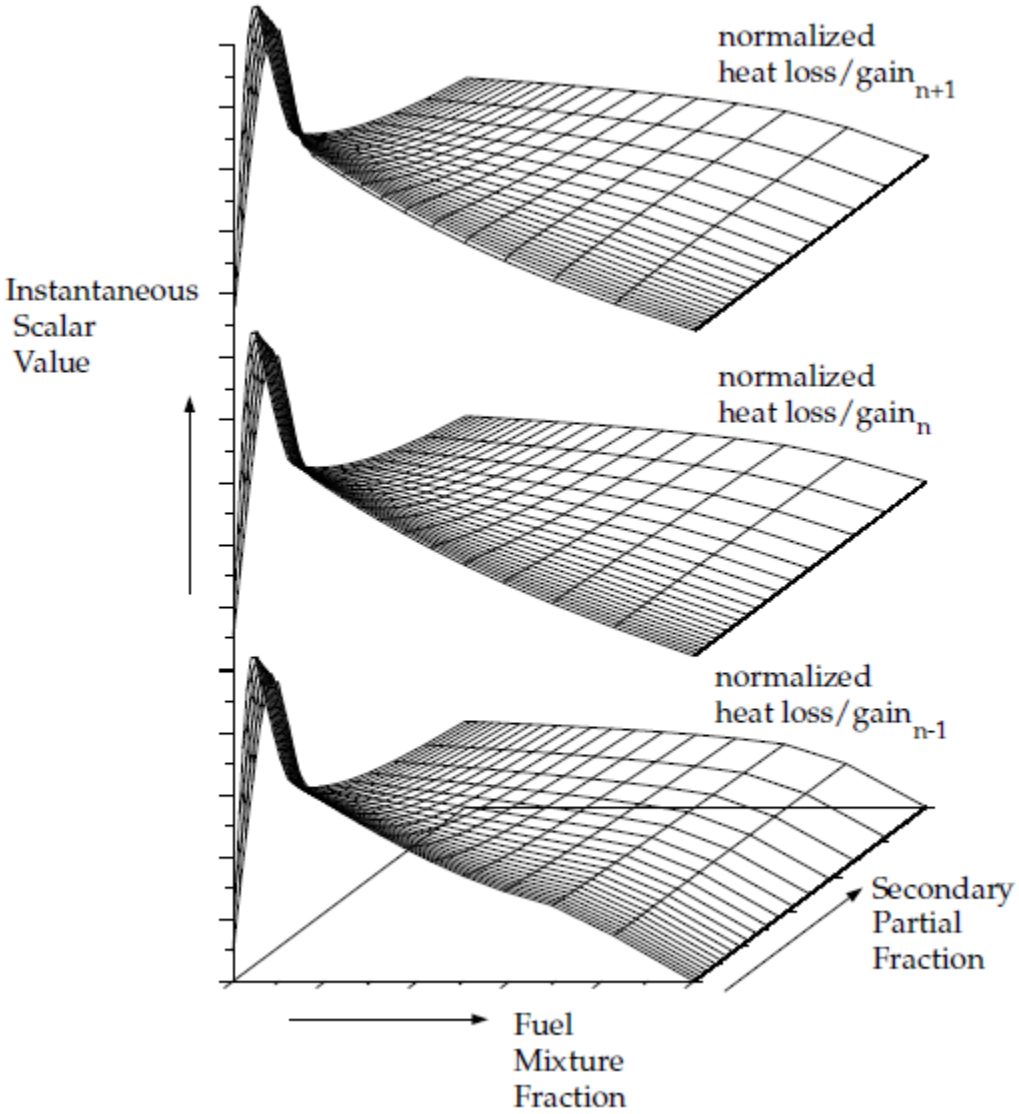
Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

especially true for cases with a large number of cells that require many iterations or time steps to converge.

Important:

Note that the computation time in Ansys Fluent for a two-mixture-fraction case will be much greater than for a single-mixture-fraction problem. This expense should be carefully considered before choosing the two-mixture-fraction model. Also, it is usually expedient to start a two-mixture-fraction simulation from a converged single-mixture-fraction solution.

Figure 8.11: Visual Representation of a Look-Up Table for the Scalar ϕ_I as a Function of Fuel Mixture Fraction and Secondary Partial Fraction, and Normalized Heat Loss/Gain in Non-Adiabatic Two-Mixture-Fraction Systems



8.1.2.4.3. Generating Lookup Tables Through Automated Grid Refinement

Ansys Fluent has the capability of generating lookup tables using **Automated Grid Refinement**. The adaptive algorithm inserts grid points in all table dimensions so that changes in values of tabulated variables (such as mean temperature, density, and species mass fractions) between successive grid points as well as changes in their slopes are less than a user-specified tolerance. The advantage of automated grid refinement is that tabulated quantities are resolved only in regions of rapid change. Therefore, compared to a fixed grid, more accurate and/or smaller PDF tables will be generated.

When using automated grid refinement, table points are calculated on a coarse grid with a user specified **Initial Number of Grid Points** (default of 15). A new grid point is inserted midway between a point i and its neighbor $i+1$ if,

$$\left|v_i - v_{i+1}\right| > \varepsilon_v\left(v_{max} - v_{min}\right) \tag{8.28}$$

where v_i is the value of a table variable at grid point i, ε_v is a user specified **Maximum Change in Value Ratio** (default of 0.25), and v_{max} (v_{min}) are the maximum (minimum) table values over all grid points.

In addition to ensuring gradual change in value between successive table points, a grid point is added if

$$\left|s_i - s_{i+1}\right| > \varepsilon_s\left(s_{max} - s_{min}\right) \tag{8.29}$$

where the slope at any point s_i is defined as,

$$s_i = \frac{v_{i+1} - v_i}{\varphi_{i+1} - \varphi_i} \tag{8.30}$$

In Equation 8.29 (p. 312) and Equation 8.30 (p. 312), ε_s is a user specified **Maximum Change in Slope Ratio** (default of 0.25), s_{max} (s_{min}) are the maximum (minimum) slopes over all grid points, and φ_i is the value of the independent grid variable being refined (that is, mean mixture fraction, mixture fraction variance or mean enthalpy).

The automated grid refinement algorithm is summarized as follows: an initial grid in the mean mixture fraction dimension is created with the specified **Initial Number of Grid Points**. Grid points are then inserted when the change in table mean temperature, and mean H2, CO and OH mole fractions, between two grid points, or the change in their slope between three grid points, is more than what you specified in **Maximum Change in Value/Slope Ratio**. This process is repeated until all points satisfy the specified value and slope change requirements, or the **Maximum Number of Grid Points** is exceeded. This procedure is then repeated for the mixture fraction variance dimension, which is calculated at the mean stoichiometric mixture fraction. Finally, if the non-adiabatic option is enabled, the procedure is repeated for the mean enthalpy grid, evaluated at the mean stoichiometric mixture fraction and zero mixture fraction variance.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

When steady diffusion flamelets are used, the **Automated Grid Refinement** can generate different mean mixture fraction grid points than specified in the diffusion flamelets. For such cases, a 4th order polynomial interpolation is used to obtain the flamelet solution at the inserted points.

Note:

The second order interpolation option is faster than fourth order interpolation, especially for tables with high dimensions, such as non-adiabatic flamelets. However, under-resolved tables that are interpolated with second order can lead to convergence difficulties. Therefore, the Automated Grid Refinement (AGR) option places grid points optimally when generating PDF tables. Hence, AGR should be used in general, and especially for second order interpolation.

Note:

Automated grid refinement is not available with two mixture fractions.

For more information about the input parameters used to generate lookup tables, refer to Calculating the Look-Up Tables in the User's Guide.

8.1.3. Restrictions and Special Cases for Using the Non-Premixed Model

For information about restrictions and special cases for using the non-premixed model, see the following sections:

8.1.3.1. Restrictions on the Mixture Fraction Approach

8.1.3.2. Using the Non-Premixed Model for Liquid Fuel or Coal Combustion

8.1.3.3. Using the Non-Premixed Model with Flue Gas Recycle

8.1.3.4. Using the Non-Premixed Model with the Inert Model

8.1.3.1. Restrictions on the Mixture Fraction Approach

The unique dependence of ϕ_i (species mass fractions, density, or temperature) on f (Equation 8.12 (p. 300) or Equation 8.14 (p. 300)) requires that the reacting system meet the following conditions:

- The chemical system must be of the diffusion type with discrete fuel and oxidizer inlets (spray combustion and pulverized fuel flames may also fall into this category).
- The Lewis number must be unity. (This implies that the diffusion coefficients for all species and enthalpy are equal, a good approximation in turbulent flow).
- When a single mixture fraction is used, the following conditions must be met:
  - Only one type of fuel is involved. The fuel may be made up of a burnt mixture of reacting species (for example, 90% CH_4 and 10% CO) and you may include multiple fuel inlets. The multiple fuel inlets must have the same composition; two or more fuel inlets with different fuel composition are not allowed (for example, one inlet of CH_4 and one inlet of CO). Similarly,

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

in spray combustion systems or in systems involving reacting particles, only one off-gas is permitted.

  - Only one type of oxidizer is involved. The oxidizer may consist of a mixture of species (for example, 21% O_2 and 79% N_2) and you may have multiple oxidizer inlets. The multiple oxidizer inlets must, however, have the same composition. Two or more oxidizer inlets with different compositions are not allowed (for example, one inlet of air and a second inlet of pure oxygen).

- When two mixture fractions are used, three streams can be involved in the system. Valid systems are as follows:

  - Two fuel streams with different compositions and one oxidizer stream. Each fuel stream may be made up of a mixture of reacting species (for example, 90% CH_4 and 10% CO). You may include multiple inlets of each fuel stream, but each fuel inlet must have one of the two defined compositions (for example, one inlet of CH_4 and one inlet of CO).

  - Mixed fuel systems including gas-liquid, gas-coal, or liquid-coal fuel mixtures with a single oxidizer. In systems with a gas-coal or liquid-coal fuel mixture, the coal volatiles and char can be treated as a single composite fuel stream and the secondary stream can represent another fuel. Alternatively, for coal combustion, the volatile and char off-gases are tracked separately as distinct fuel streams.

  - Two oxidizer streams with different compositions and one fuel stream. Each oxidizer stream may consist of a mixture of species (for example 21% O_2 and 79% N_2). You may have multiple inlets of each oxidizer stream, but each oxidizer inlet must have one of the two defined compositions (for example, one inlet of air and a second inlet of pure oxygen).

  - A fuel stream, an oxidizer stream, and a non-reacting secondary stream.

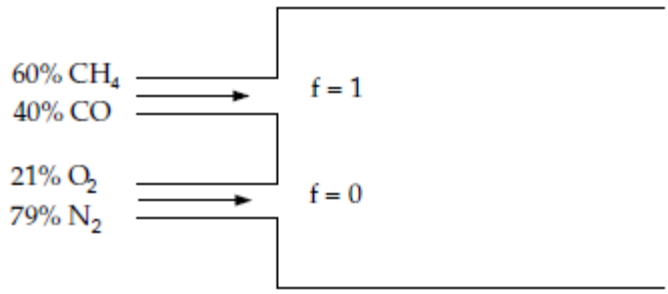
- The flow must be turbulent.

It is important to emphasize that these restrictions eliminate the use of the non-premixed approach for directly modeling premixed combustion. This is because the unburned premixed stream is far from chemical equilibrium. Note, however, that an extended mixture fraction formulation, the partially premixed model (see Partially Premixed Combustion (p. 343)), can be applied to non-premixed (with mixed-but-unburnt regions), as well as partially premixed flames.

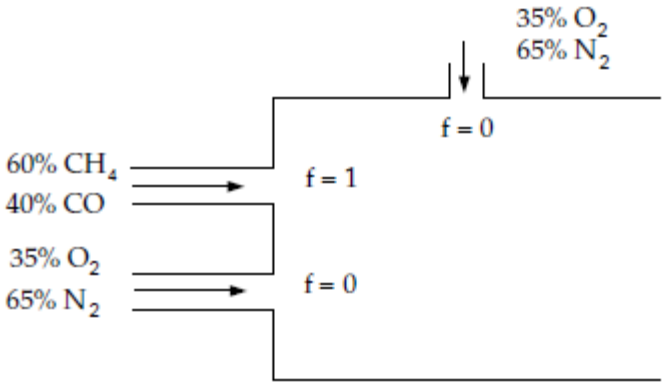
Figure 8.12: Chemical Systems That Can Be Modeled Using a Single Mixture Fraction (p. 315) and Figure 8.13: Chemical System Configurations That Can Be Modeled Using Two Mixture Fractions (p. 316) illustrate typical reacting system configurations that can be handled by the non-premixed model in Ansys Fluent. Figure 8.14: Premixed Systems That Cannot Be Modeled Using the Non-Premixed Model (p. 316) shows a premixed configuration that cannot be modeled using the non-premixed model.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

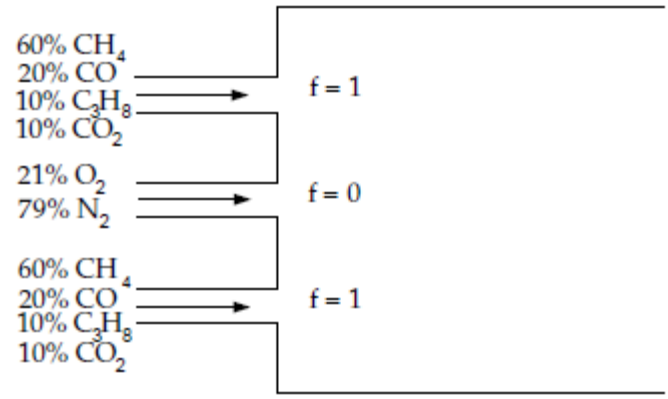
Figure 8.12: Chemical Systems That Can Be Modeled Using a Single Mixture Fraction



(a) Simple Fuel/Oxidant Diffusion Flame



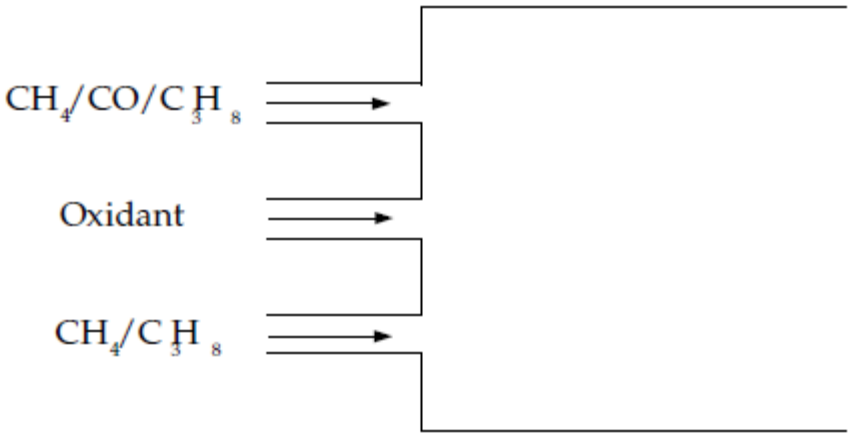
(b) Diffusion System Using Multiple Oxidant Inlets



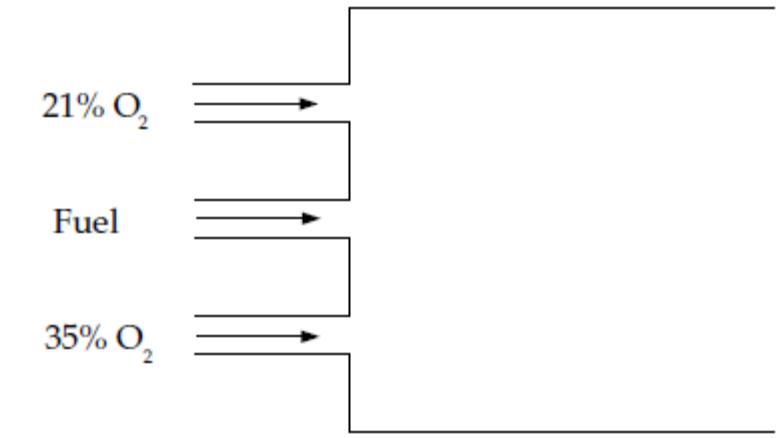
(c) System Using Multiple Fuel Inlets

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Figure 8.13: Chemical System Configurations That Can Be Modeled Using Two Mixture Fractions

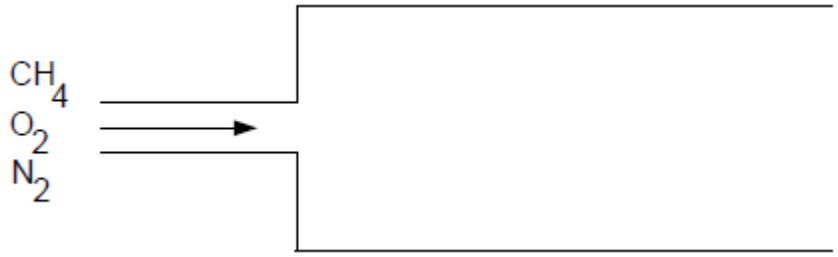


(a) System Containing Two Dissimilar Fuel Inlets



(b) System Containing Two Dissimilar Oxidant Inlets

Figure 8.14: Premixed Systems That Cannot Be Modeled Using the Non-Premixed Model



8.1.3.2. Using the Non-Premixed Model for Liquid Fuel or Coal Combustion

You can use the non-premixed model if your Ansys Fluent simulation includes liquid droplets and/or coal particles. In this case, fuel enters the gas phase within the computational domain at a rate determined by the evaporation, devolatilization, and char combustion laws governing the dispersed

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

phase. In the case of coal, the volatiles and the products of char can be defined as two different types of fuel (using two mixture fractions) or as a single composite off-gas (using one mixture fraction), as described in Modeling Coal Combustion Using the Non-Premixed Model in the User's Guide.

8.1.3.3. Using the Non-Premixed Model with Flue Gas Recycle

While most problems you solve using the non-premixed model will involve inlets that contain either pure oxidant or pure fuel ($f=0$ or 1), you can include an inlet that has an intermediate value of mixture fraction ($0<f<1$) provided that this inlet represents a completely reacted mixture. Such cases arise when there is flue gas recirculation, as depicted schematically in Figure 8.15: Using the Non-Premixed Model with Flue Gas Recycle (p. 317). Since f is a conserved quantity, the mixture fraction at the flue gas recycle inlet can be computed as

$$\dot{m}_{fuel}+\dot{m}_{recyc}f_{exit}=\left(\dot{m}_{fuel}+\dot{m}_{ox}+\dot{m}_{recyc}\right)f_{exit} \tag{8.31}$$

or

$$f_{exit}=\frac{\dot{m}_{fuel}}{\dot{m}_{fuel}+\dot{m}_{ox}} \tag{8.32}$$

where f_{exit} is the exit mixture fraction (and the mixture fraction at the flue gas recycle inlet), $\dot{m}_{ox}$ is the mass flow rate of the oxidizer inlet, $\dot{m}_{fuel}$ is the mass flow rate of the fuel inlet, $\dot{m}_{recyc}$ is the mass flow rate of the recycle inlet.

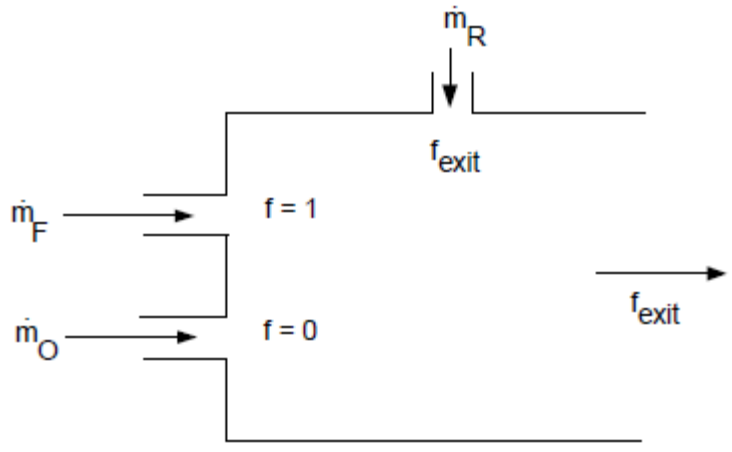
If a secondary stream is included,

$$f_{fuel,exit}=\frac{\dot{m}_{fuel}}{\dot{m}_{fuel}+\dot{m}_{sec}+\dot{m}_{ox}} \tag{8.33}$$

and

$$p_{sec,exit}=\frac{\dot{m}_{sec}}{\dot{m}_{sec}+\dot{m}_{ox}} \tag{8.34}$$

Figure 8.15: Using the Non-Premixed Model with Flue Gas Recycle



8.1.3.4. Using the Non-Premixed Model with the Inert Model

To model the effect of dilution on combustion without the expense of using two mixture fractions, Ansys Fluent allows the introduction of an inert stream into the domain. Unlike a secondary mixture fraction, the inert does not chemically equilibrate with the primary fuel and oxidizer - instead, its composition remains constant after mixing. However the inert stream does affect the solution due to its influence on enthalpy, specific heat, and density of the mixture. The equation for conservation of inert is written as:

$$\frac{\partial \rho Y_I}{\partial t} + \nabla \cdot \left(\vec{v} \rho Y_I \right) = \nabla \cdot \left(\frac{\mu_l + \mu_t}{Sc_t} \nabla \left(Y_I \right) \right) \tag{8.35}$$

where

Y_I = inert stream tracer

Sc_t = turbulent Schmidt number

μ_t = turbulent viscosity

ρ = density

Equation 8.35 (p. 318) has no sources or sinks, because the problem is reduced to tracking a conserved scalar when it is assumed that the inert components have the same turbulent diffusivities.

8.1.3.4.1. Mixture Composition

The mixture properties are computed from the mean ($\overline{f}$) and variance ($\overline{f'}$) of the mixture fraction in the cell, the reaction progress variable (c, when the partially premixed model is enabled), the cell enthalpy (H, for non-adiabatic flows), and the inert tracer (Y_I). The mixture is modeled as a blend of inert and active species, but the PDF tables need to be accessed with *conditioned* variables. Conditioning is necessary to take into account the volume taken up by the inert fraction, yet still be able to use previously built tables by straightforward lookup. The mean mixture fraction and mixture fraction variance used to access the PDF table is given by:

$$\overline{f}_I = \frac{\overline{f}}{1 - Y_I} \tag{8.36}$$

$$f_I'^2 = \frac{\overline{f'^2}}{\left(1 - Y_I\right)^2} \tag{8.37}$$

The reaction progress variable c is not conditioned, however the cell enthalpy must be conditioned to account for the inert enthalpy. The inert enthalpy and active enthalpy are obtained from the following relationships:

$$H(T) = \left(1 - Y_I\right) H^{pdf}(T) + Y_I H^I(T) \tag{8.38}$$

where H is the enthalpy of the cell at temperature T, H^{pdf} is the enthalpy of the active mixture-fraction stream and H^I is the enthalpy of the inert stream. Here it is assumed that the inert and the active streams have the same temperature, but different enthalpies. To calculate the temperature in the cell, Equation 8.38 (p. 318) is solved for the temperature and for H^{pdf}, which gives the partitioning of the energy between the inert and active streams.

The inert enthalpy is defined as

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$H^I(T) = \sum_{i=0}^{N_I} Y_i \left(h_i^0 + \int_{T_0}^{T} C_{p_i} dT \right) \tag{8.39}$$

where Y_i refers to the mass fraction of species i defined in the inert stream, h_i^0 is the formation enthalpy of species i, T_0 is the reference temperature, C_{p_i} the specific heat of species i, and N_I is the number of inert species.

The inert and PDF enthalpies are defined further in Equation 41.11 in the User's Guide.

8.1.3.4.1.1. Property Evaluation

The specific heat of the mixture is evaluated by mixing the inert and active streams in the following way:

$$C_p(T) = (1 - Y_I) C_p^{pdf}(T) + Y_I C_p^I(T) \tag{8.40}$$

The density of the mixture is calculated by using a harmonic average of the densities of the active and inert streams, weighted by the inert tracer:

$$\rho(T) = \left[\frac{(1 - Y_I)}{\rho^{pdf}(T)} + \frac{Y_I}{\rho^I(T)} \right]^{-1} \tag{8.41}$$

Here, the inert density (ρ^I) is calculated from the ideal gas law.

For information on how to set up the inert model, see Setting Up the Inert Model in the User's Guide.

8.1.4. The Diffusion Flamelet Models Theory

Information about the flamelet models are presented in the following sections:

8.1.4.1. Restrictions and Assumptions
8.1.4.2. The Flamelet Concept
8.1.4.3. Flamelet Generation
8.1.4.4. Flamelet Import

8.1.4.1. Restrictions and Assumptions

The following restrictions apply to all diffusion flamelet models in Ansys Fluent:

- Only a single mixture fraction can be modeled; two-mixture-fraction flamelet models are not allowed.
- The mixture fraction is assumed to follow the β-function PDF, and scalar dissipation fluctuations are ignored.
- Empirically-based streams cannot be used with the flamelet model.

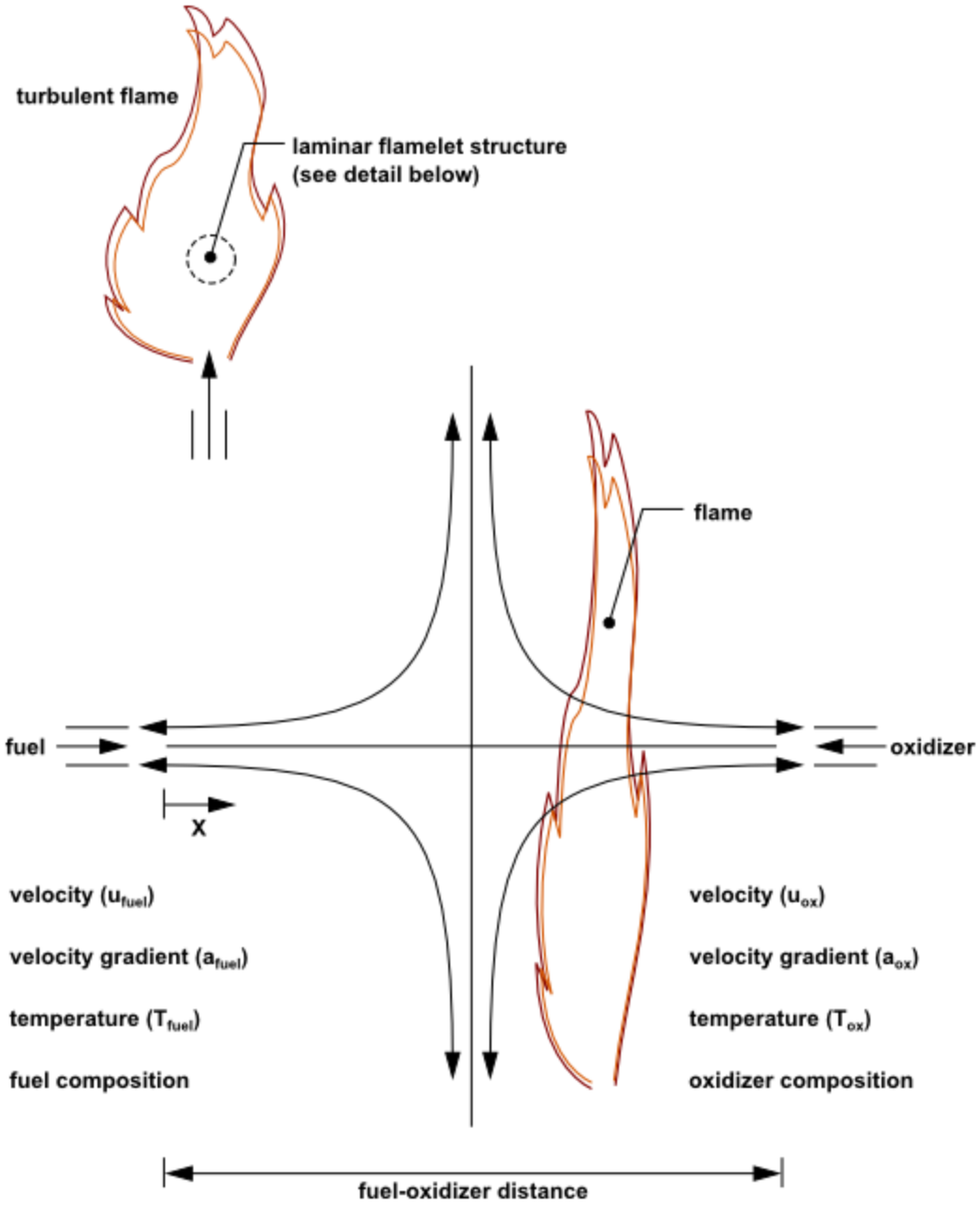
Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

8.1.4.2. The Flamelet Concept

8.1.4.2.1. Overview

The flamelet concept views the turbulent flame as an ensemble of thin, laminar, locally one-dimensional flamelet structures embedded within the turbulent flow field [77] (p. 1057), [509] (p. 1082), [510] (p. 1082) (see Figure 8.16: Laminar Opposed-Flow Diffusion Flamelet (p. 320)).

Figure 8.16: Laminar Opposed-Flow Diffusion Flamelet



A common laminar flame type used to represent a flamelet in a turbulent flow is the counterflow diffusion flame. This geometry consists of opposed, axisymmetric fuel and oxidizer jets. As the distance between the jets is decreased and/or the velocity of the jets increased, the flame is strained and increasingly departs from chemical equilibrium until it is eventually extinguished.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

The species mass fraction and temperature fields can be measured in laminar counterflow diffusion flame experiments, or, most commonly, calculated. For the latter, a self-similar solution exists, and the governing equations can be simplified to one dimension along the axis of the fuel and oxidizer jets, where complex chemistry calculations can be affordably performed.

In the laminar counterflow flame, the mixture fraction, f, (see Definition of the Mixture Fraction (p. 296)) decreases monotonically from unity at the fuel jet to zero at the oxidizer jet. If the species mass fraction and temperature along the axis are mapped from physical space to mixture fraction space, they can be uniquely described by two parameters: the mixture fraction and the strain rate (or, equivalently, the scalar dissipation, χ, defined in Equation 8.43 (p. 321)). Hence, the chemistry is reduced and completely described by the two quantities, f and χ.

This reduction of the complex chemistry to two variables allows the flamelet calculations to be preprocessed, and stored in look-up tables. By preprocessing the chemistry, computational costs are reduced considerably.

The balance equations, solution methods, and sample calculations of the counterflow laminar diffusion flame can be found in several references. Comprehensive reviews and analyses are presented in the works of Bray and Peters, and Dixon-Lewis [77] (p. 1057), [143] (p. 1061).

8.1.4.2.2. Strain Rate and Scalar Dissipation

A characteristic strain rate for a counterflow diffusion flamelet can be defined as $a_s = v / d$, where v is the relative speed of the fuel and oxidizer jets, and d is the distance between the jet nozzles.

Instead of using the strain rate to quantify the departure from equilibrium, it is expedient to use the scalar dissipation, denoted by χ. The scalar dissipation is defined as

$$\chi = 2D\left|\nabla f\right|^2 \tag{8.42}$$

where D is a representative diffusion coefficient.

Note that the scalar dissipation, χ, varies along the axis of the flamelet. For the counterflow geometry, the flamelet strain rate a_s can be related to the scalar dissipation at the position where f is stoichiometric by [509] (p. 1082):

$$\chi_{st} = \frac{a_s \exp\left(-2\left[erfc^{-1}\left(2f_{st}\right)\right]^2\right)}{\pi} \tag{8.43}$$

where

χ_{st} = scalar dissipation at $f = f_{st}$

a_s = characteristic strain rate

f_{st} = stoichiometric mixture fraction

$erfc^{-1}$ = inverse complementary error function

Physically, as the flame is strained, the width of the reaction zone diminishes, and the gradient of f at the stoichiometric position $f = f_{st}$ increases. The instantaneous stoichiometric scalar dissipation, χ_{st}, is used as the essential non-equilibrium parameter. It has the dimensions s^{-1} and may be interpreted as the inverse of a characteristic diffusion time. In the limit $\chi_{st} \to 0$ the

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

chemistry tends to equilibrium, and as χ_{st} increases due to aerodynamic straining, the non-equilibrium increases. Local quenching of the flamelet occurs when χ_{st} exceeds a critical value.

8.1.4.2.3. Embedding Diffusion Flamelets in Turbulent Flames

A turbulent flame brush is modeled as an ensemble of discrete diffusion flamelets. Since, for adiabatic systems, the species mass fraction and temperature in the diffusion flamelets are completely parameterized by f and χ_{st}, density-weighted mean species mass fractions and temperature in the turbulent flame can be determined from the PDF of f and χ_{st} as:

$$\overline{\phi}=\int\int \phi\left(f,\chi_{st}\right)p\left(f,\chi_{st}\right)df d\chi_{st} \tag{8.44}$$

where ϕ represents species mass fractions and temperature.

In Ansys Fluent, f and χ_{st} are assumed to be statistically independent, so the joint PDF $p\left(f,\chi_{st}\right)$ can be simplified as $p_f(f)p_\chi\left(\chi_{st}\right)$. A β PDF shape is assumed for p_f, and transport equations for $\overline{f}$ and $\overline{f'^2}$ are solved in Ansys Fluent to specify p_f. Fluctuations in χ_{st} are ignored so that the PDF of χ is a delta function: $p_\chi=\delta\left(\chi-\overline{\chi_{st}}\right)$. The first moment, namely the mean scalar dissipation, $\overline{\chi_{st}}$, is modeled in Ansys Fluent as

$$\overline{\chi_{st}}=\frac{C_\chi\overline{f'^2}}{\tau} \tag{8.45}$$

where C_χ is a constant with a default value of 2, and τ is the turbulent time scale calculated according to the turbulence model. For example, for RANS turbulence models,

$$\tau=\frac{k}{\varepsilon} \tag{8.46}$$

For non-adiabatic steady diffusion flamelets, the additional parameter of enthalpy is required. However, the computational cost of modeling steady diffusion flamelets over a range of enthalpies is prohibitive, so some approximations are made. Heat gain/loss to the system is assumed to have a negligible effect on the species mass fractions, and adiabatic mass fractions are used [62] (p. 1056), [461] (p. 1079). The temperature is then calculated from Equation 5.8 (p. 173) for a range of mean enthalpy gain/loss, $\overline{H}$. Accordingly, mean temperature and density PDF tables have an extra dimension of mean enthalpy. The approximation of constant adiabatic species mass fractions is, however, not applied for the case corresponding to a scalar dissipation of zero. Such a case is represented by the non-adiabatic equilibrium solution. For $\overline{\chi_{st}}=0$, the species mass fractions are computed as functions of $\overline{f}$, $\overline{f'^2}$, and $\overline{H}$.

In Ansys Fluent, you can either generate your own diffusion flamelets, or import them as flamelet files calculated with other stand-alone packages. Such stand-alone codes include OPPDIF [398] (p. 1076), CFX-RIF [45] (p. 1055), [46] (p. 1055), [519] (p. 1083) and RUN-1DL [513] (p. 1083). Ansys Fluent can import flamelet files in standard flamelet file format.

Instructions for generating and importing diffusion flamelets are provided in Flamelet Generation (p. 323) and Flamelet Import (p. 323).

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

8.1.4.3. Flamelet Generation

The laminar counterflow diffusion flame equations can be transformed from physical space (with x as the independent variable) to mixture fraction space (with f as the independent variable) [520] (p. 1083). In Ansys Fluent, a simplified set of the mixture fraction space equations are solved [519] (p. 1083). Here, N equations are solved for the species mass fractions, Y_i,

$$\rho\frac{\partial Y_i}{\partial t}=\frac{1}{2}\rho\chi\frac{\partial^2 Y_i}{\partial f^2}+S_i \tag{8.47}$$

and one equation for temperature:

$$\rho\frac{\partial T}{\partial t}=\frac{1}{2}\rho\chi\frac{\partial^2 T}{\partial f^2}-\frac{1}{c_p}\sum_i H_i S_i+\frac{1}{2c_p}\rho\chi\left[\frac{\partial c_p}{\partial f}+\sum_i c_{p,i}\frac{\partial Y_i}{\partial f}\right]\frac{\partial T}{\partial f} \tag{8.48}$$

The notation in Equation 8.47 (p. 323) and Equation 8.48 (p. 323) is as follows: Y_i, T, ρ, and f are the i^{th} species mass fraction, temperature, density, and mixture fraction, respectively. $c_{p,i}$ and c_p are the i^{th} species specific heat and mixture-averaged specific heat, respectively. S_i is the i^{th} species reaction rate, and H_i is the specific enthalpy of the i^{th} species.

The scalar dissipation, χ, must be modeled across the flamelet. An extension of Equation 8.43 (p. 321) to variable density is used [302] (p. 1070):

$$\chi(f)=\frac{a_s}{4\pi}\frac{3\left(\sqrt{\rho_\infty/\rho}+1\right)^2}{2\sqrt{\rho_\infty/\rho}+1}\exp\left(-2\left[erfc^{-1}(2f)\right]^2\right) \tag{8.49}$$

where ρ_∞ is the density of the oxidizer stream.

Using the definition of the strain rate given by Equation 8.49 (p. 323), the scalar dissipation in the mixture fraction space can be expressed as:

$$\chi(f)=\chi_{st}\frac{\phi(f,\rho)}{\phi\left(f_{st},\rho_{st}\right)} \tag{8.50}$$

where f_{st} is the stoichiometric mixture fraction, ρ_{st} is the mixture density, and χ_{st} is the scalar dissipation at $f=f_{st}$. χ_{st} is the user input for flamelet generation. $\phi(f,\rho)$ and $\phi(f_{st},\rho_{st})$ are evaluated using Equation 8.49 (p. 323).

8.1.4.4. Flamelet Import

Ansys Fluent can import one or more flamelet files, convolute these diffusion flamelets with the assumed-shape PDFs (see Equation 8.44 (p. 322)), and construct look-up tables. The flamelet files can be generated in Ansys Fluent, or with separate stand-alone computer codes.

The following types of flamelet files can be imported into Ansys Fluent:

- ASCII files generated by the CFX-RIF code [45] (p. 1055), [46] (p. 1055), [519] (p. 1083)
- Standard format files described in Standard Files for Diffusion Flamelet Modeling in the User's Guide and in Peters and Rogg [513] (p. 1083)

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

When diffusion flamelets are generated in physical space, the species and temperature vary in one spatial dimension. The species and temperature must then be mapped from physical space to mixture fraction space. If the diffusion coefficients of all species are equal, a unique definition of the mixture fraction exists. However, with differential diffusion, the mixture fraction can be defined in a number of ways.

Ansys Fluent computes the mixture fraction profile along the diffusion flamelet in one of the following ways:

- Read from a file (standard format files only)

  This option is for diffusion flamelets solved in mixture fraction space. If you choose this method, Ansys Fluent will search for the mixture fraction keyword Z, as specified in Peter and Roggs's work [513] (p. 1083), and retrieve the data. If Ansys Fluent does not find mixture fraction data in the flamelet file, it will instead use the hydrocarbon formula method described below.

- Hydrocarbon formula

  Following the work of Bilger et al. [61] (p. 1056), the mixture fraction is calculated as

$$f = \frac{b - b_{ox}}{b_{fuel} - b_{ox}} \tag{8.51}$$

  where

$$b = 2\frac{Y_C}{M_{w,C}} + 0.5\frac{Y_H}{M_{w,H}} - \frac{Y_O}{M_{w,O}} \tag{8.52}$$

  Y_C, Y_H, and Y_O are the mass fractions of carbon, hydrogen, and oxygen atoms, and $M_{w,C}$, $M_{w,H}$, and $M_{w,O}$ are the molecular weights. b_{ox} and b_{fuel} are the values of b at the oxidizer and fuel inlets.

The flamelet profiles in the multiple-flamelet data set should vary only in the strain rate imposed; the species and the boundary conditions should be the same. In addition, it is recommended that an extinguished flamelet is excluded from the multiple-flamelet data set. The formats for multiple diffusion flamelets are as follows:

- Standard format: If you have a set of standard format flamelet files, you can import them all at the same time, and Ansys Fluent will merge them internally into a multiple-flamelet file. When you import the set of flamelet files, Ansys Fluent will search for and count the occurrences of the HEADER keyword to determine the number of diffusion flamelets in the file.

- CFX-RIF format: A CFX-RIF flamelet file contains multiple diffusion flamelets at various strains and the file should not be modified manually. Only one CFX-RIF flamelet file should be imported.

For either type of file, Ansys Fluent will determine the number of flamelet profiles and sort them in ascending strain-rate order. For diffusion flamelets generated in physical space, you can select one of the four methods available for the calculation of mixture fraction. The scalar dissipation will be calculated from the strain rate using Equation 8.43 (p. 321).

8.1.5. The Steady Diffusion Flamelet Model Theory

The steady flamelet approach models a turbulent flame brush as an ensemble of discrete, steady laminar flames, called diffusion flamelets. The individual diffusion flamelets are assumed to have the

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

same structure as laminar flames in simple configurations, and are obtained by experiments or calculations. Using detailed chemical mechanisms, Ansys Fluent can calculate laminar opposed-flow diffusion flamelets for non-premixed combustion. The diffusion flamelets are then embedded in a turbulent flame using statistical PDF methods.

The advantage of the diffusion flamelet approach is that realistic chemical kinetic effects can be incorporated into turbulent flames. The chemistry can then be preprocessed and tabulated, offering tremendous computational savings. However, the steady diffusion flamelet model is limited to modeling combustion with relatively fast chemistry. The flame is assumed to respond instantaneously to the aerodynamic strain, and therefore the model cannot capture deep non-equilibrium effects such as ignition, extinction, and slow chemistry (like NOx).

Information pertaining strictly to the steady diffusion flamelet model is presented in the following sections:

8.1.5.1. Overview

8.1.5.2. Multiple Steady Flamelet Libraries

8.1.5.3. Steady Diffusion Flamelet Automated Grid Refinement

8.1.5.4. Non-Adiabatic Steady Diffusion Flamelets

For general information about the mixture fraction model, see Introduction (p. 295).

8.1.5.1. Overview

In a diffusion flame, at the molecular level, fuel and oxidizer diffuse into the reaction zone. Here, they encounter high temperatures and radical species and ignite. More heat and radicals are generated in the reaction zone and some diffuse out. In near-equilibrium flames, the reaction rate is much faster than the diffusion rate. However, as the flame is stretched and strained by the turbulence, species and temperature gradients increase, and radicals and heat diffuse more quickly out of the flame. The species have less time to reach chemical equilibrium, and the degree of local non-equilibrium increases.

The steady diffusion flamelet model is suited to predict chemical non-equilibrium due to aerodynamic straining of the flame by the turbulence. The chemistry, however, is assumed to respond rapidly to this strain, so as the strain relaxes to zero, the chemistry tends to equilibrium.

When the chemical time-scale is comparable to the fluid mixing time-scale, the species can be considered to be in global chemical non-equilibrium. Such cases include NOx formation and low-temperature CO oxidation. The steady diffusion flamelet model is not suitable for such slow-chemistry flames. Instead, you can model slow chemistry using one of the following:

- the Unsteady Diffusion Flamelet model (see The Unsteady Diffusion Flamelet Model Theory (p. 327))
- the trace species assumption in the NOx model (see Pollutant Formation (p. 363))
- the Laminar Finite-Rate model (see The Generalized Finite-Rate Formulation for Reaction Modeling (p. 243)), where the turbulence-chemistry interaction is ignored.
- the EDC model (see The Eddy-Dissipation-Concept (EDC) Model (p. 249))
- the PDF transport model (see Composition PDF Transport (p. 276)).

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

8.1.5.2. Multiple Steady Flamelet Libraries

Ansys Fluent can generate multiple steady diffusion flamelets over a range of strain rates to account for the varying strain field in your multi-dimensional simulation. If you specify the number of diffusion flamelets to be greater than one, diffusion flamelets are generated at scalar dissipation values as determined by Equation 8.53 (p. 326).

$$\chi_i = \begin{cases} 10\chi_{i-1} & for\ \chi_{i-1} < 1s^{-1} \\ \chi_{i-1} + \Delta\chi & for\ \chi_{i-1} \geq 1s^{-1} \end{cases} \tag{8.53}$$

where i ranges from 1 up to the specified maximum number of diffusion flamelets, χ_0 is the initial scalar dissipation at the stoichiometric mixture fraction χ_{st} from Equation 8.50 (p. 323), and $\Delta\chi$ is the scalar dissipation step. Diffusion flamelets are generated until either the maximum number of flamelets is reached, or the flamelet extinguishes. Extinguished flamelets are excluded from the flamelet library.

8.1.5.3. Steady Diffusion Flamelet Automated Grid Refinement

By default, 1D flamelet grids are discretized by clustering a fixed number of points about the stoichiometric mixture fraction, which is approximated as the location of peak temperature. Ansys Fluent also has the option for **Automated Grid Refinement** of steady diffusion flamelets, where an adaptive algorithm inserts grid points so that the change of values, as well as the change of slopes, between successive grid points is less than user specified tolerances.

When using automated grid refinement, a steady solution is calculated on a coarse grid with a user specified **Initial Number of Grid Points in Flamelet** (default of 8). At convergence, a new grid point is inserted midway between a point i and its neighbor $i+1$ if,

$$\left|v_i - v_{i+1}\right| > \varepsilon_v\left(v_{max} - v_{min}\right) \tag{8.54}$$

where v_i is the value for each temperature and species mass fraction at grid point i, ε_v is a user specified **Maximum Change in Value Ratio** (default of 0.5), and v_{max} (v_{min}) are the maximum (minimum) values over all grid points.

In addition a grid point is added if,

$$\left|s_i - s_{i+1}\right| > \varepsilon_s\left(s_{max} - s_{min}\right) \tag{8.55}$$

where the slope s_i is defined as,

$$s_i = \frac{v_{i+1} - v_i}{f_{i+1} - f_i} \tag{8.56}$$

In Equation 8.55 (p. 326) and Equation 8.56 (p. 326), ε_s is a user specified **Maximum Change in Slope Ratio** (default of 0.5), s_{max} (s_{min}) are the maximum (minimum) slopes over all grid points, and f_i is the mixture fraction value of grid point i.

The refined flamelet is reconverged, and the refinement process is repeated until no further grid points are added by Equation 8.54 (p. 326) and Equation 8.55 (p. 326), or the user-specified **Maximum Number of Grid Points in Flamelet** (default of 64) is exceeded.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

8.1.5.4. Non-Adiabatic Steady Diffusion Flamelets

For non-adiabatic steady diffusion flamelets, Ansys Fluent follows the approach of [62] (p. 1056), [461] (p. 1079) and assumes that flamelet species profiles are unaffected by heat loss/gain from the flamelet. No special non-adiabatic flamelet profiles need to be generated, avoiding a very cumbersome preprocessing step. In addition, the compatibility of Ansys Fluent with external steady diffusion flamelet generation packages (for example, OPPDIF, CFX-RIF, RUN-1DL) is retained. The disadvantage to this model is that the effect of the heat losses on the species mass fractions is not taken into account. Also, the effect of the heat loss on the extinction limits is not taken into account.

After diffusion flamelet generation, the flamelet profiles are convoluted with the assumed-shape PDFs as in Equation 8.44 (p. 322), and then tabulated for look-up in Ansys Fluent. The non-adiabatic PDF tables have the following dimensions:

$\overline{T}\left(\overline{f},\overline{f'^2},\overline{H},\overline{\chi_{st}}\right)$

$\overline{Y_i}\left(\overline{f},\overline{f'^2},\overline{H}\right)$ for $\chi = 0$ (that is, equilibrium solution)

$\overline{Y_i}\left(\overline{f},\overline{f'^2},\overline{\chi_{st}}\right)$ for $\chi \neq 0$

$\overline{\rho}\left(\overline{f},\overline{f'^2},\overline{H},\overline{\chi_{st}}\right)$

During the Ansys Fluent solution, the equations for the mean mixture fraction, mixture fraction variance, and mean enthalpy are solved. The scalar dissipation field is calculated from the turbulence field and the mixture fraction variance (Equation 8.45 (p. 322)). The mean values of cell temperature, density, and species mass fraction are obtained from the PDF look-up table.

8.1.6. The Unsteady Diffusion Flamelet Model Theory

The *steady* diffusion flamelet model, described in The Diffusion Flamelet Models Theory (p. 319) and The Steady Diffusion Flamelet Model Theory (p. 324), models local chemical non-equilibrium due to the straining effect of turbulence. In many combustors the strain is small at the outlet and the steady diffusion flamelet model predicts all species, including slow-forming species like NOx, to be near equilibrium, which is often inaccurate. The cause of this inaccuracy is the disparity between the flamelet time-scale, which is the inverse of the scalar dissipation, and the slow-forming species time-scale, which is the residence time since the species started accumulating after mixing in the combustor.

The *unsteady* diffusion flamelet model in Ansys Fluent can predict slow-forming species, such as gaseous pollutants or product yields in liquid reactors, more accurately than the steady diffusion flamelet model. Computationally expensive chemical kinetics are reduced to one dimension and the model is significantly faster than the laminar-finite-rate, EDC or PDF Transport models where kinetics are calculated in two or three dimensions. There are two variants of the unsteady flamelet model, namely an Eulerian unsteady flamelet model (described in The Eulerian Unsteady Laminar Flamelet Model (p. 328)) and a diesel unsteady flamelet model for predicting combustion in compression-ignition engines (described in The Diesel Unsteady Laminar Flamelet Model (p. 330)).

Information pertaining strictly to the unsteady flamelet model is presented in the following sections:

8.1.6.1. The Eulerian Unsteady Laminar Flamelet Model

8.1.6.2. The Diesel Unsteady Laminar Flamelet Model

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

8.1.6.3. Multiple Diesel Unsteady Flamelets
8.1.6.4. Multiple Diesel Unsteady Flamelets with Flamelet Reset

8.1.6.1. The Eulerian Unsteady Laminar Flamelet Model

The Eulerian unsteady laminar flamelet model can be used to predict slow-forming intermediate and product species that are not in chemical equilibrium. Typical examples of slow-forming species are gas-phase pollutants like NOx, and product compounds in liquid reactors. By reducing the chemistry computations to one dimension, detailed kinetics with multiple species and stiff reactions can be economically simulated in complex 3D geometries.

The model, following the work of Barths et al. [46] (p. 1055) and Coelho and Peters [112] (p. 1059), postprocesses unsteady marker probability equations on a steady-state converged flow field. The marker field represents the probability of finding a flamelet at any point in time and space. A probability marker transport equation is solved for each flamelet.

In Ansys Fluent, the steady flow solution must be computed with the steady diffusion flamelet model (see The Steady Diffusion Flamelet Model Theory (p. 324)) before starting the unsteady flamelets simulation. Since the unsteady flamelet equations are postprocessed on a steady-state flamelet solution, the effect of the unsteady flamelet species on the flow-field is neglected.

When multiple flamelets are enabled, Ansys Fluent solves Eulerian transport equations representing the probability of fuel for n^{th} flamelet, I_n, as follows

$$\frac{\partial(\rho I_n)}{\partial t}+\nabla\cdot(\rho\vec{v}I_n)=\nabla\cdot\left(\left(\frac{k}{C_p}+\frac{\mu_t}{\sigma_t}\right)\nabla I_n\right) \tag{8.57}$$

where k is laminar thermal conductivity of the mixture, C_p is the mixture specific heat, and σ_t is the Prandtl number.

Each marker probability is initialized as,

$$I_n=\begin{cases}1 & for\ \chi_{\min,n}<\chi\le\chi_{\max,n}\ and\ \overline{f}\ge f_{init}\\ 0 & for\ \chi<\chi_{\min,n}\ or\ \chi>\chi_{\max,n}\ or\ \overline{f}<f_{init}\end{cases} \tag{8.58}$$

where χ is the scalar dissipation, $\chi_{\min,n}\ and\ \chi_{\max,n}$ are the minimum and maximum scalar dissipation values for n^{th} flamelet marker probability, $\overline{f}$ is the mean mixture fraction, and f_{init} is a user-specified constant that should be set greater than the stoichiometric mixture fraction. Hence the marker probabilities are initialized to unity in regions of the domain where the mean mixture fraction is greater than a user specified value (typically greater than stoichiometric), and multiple flamelets sub-divide this initial region by the scalar dissipation.

As previously mentioned, the Eulerian Unsteady Laminar Flamelet Model is only available for steady-state simulations. However, marker probability transport equations (Equation 8.57 (p. 328)) are always solved time-accurately as the initial marker probabilities convect and diffuse through the steady flow field. At the inlet boundaries, I_n is set to zero, and hence the I_n field decreases to zero with time as it is convected and diffused out of the domain (for cases with outlet boundaries).

The unsteady flamelet species Equation 8.47 (p. 323)) is integrated simultaneously with the marker probability Equation 8.57 (p. 328) for each marker probability I_n. For liquid-phase chemistry, the initial flamelet field is the mixed-but-unburnt flamelet, as liquid reactions are assumed to proceed immediately upon mixing. Gas-phase chemistry involves ignition, so the initial flamelet field is cal-

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

culated from a steady diffusion flamelet solution. All of the slow-forming species, such as NOx, must be identified before solving the unsteady flamelet equations. The mass fractions of all slow-forming species are set to zero in this initial flamelet profile, since, at ignition, little residence time has elapsed for any significant formation.

The scalar dissipation at stoichiometric mixture fraction ($\chi_{st,n}$) is required by each flamelet species equation. This is calculated from the steady-state Ansys Fluent field at each time step as a probability-weighted volume integral:

$$\chi_{st,n}(t)=\frac{\int_V I_n\left(\overline{\chi}_{st}(\vec{x})\right)^{\frac{3}{2}}\rho(\vec{x})dV}{\int_V I_n\left(\overline{\chi}_{st}(\vec{x})\right)^{\frac{1}{2}}\rho(\vec{x})dV} \tag{8.59}$$

where $\overline{\chi_{st}}$ is defined in Equation 8.45 (p. 322), and V denotes the fluid volume. Ansys Fluent provides the option of limiting $\overline{\chi_{st}}$ to a user-specified maximum value, which should be approximately equal to the flamelet extinction scalar dissipation (the steady diffusion flamelet solver can be used to calculate this extinction scalar dissipation in a separate simulation).

The unsteady flamelet energy equation is not solved in order to avoid flamelet extinction for high scalar dissipation, and to account for non-adiabatic heat loss or gain. For adiabatic cases, the flamelet temperature, $T^{ad}\left(f,t\right)$, is calculated at each time step from the steady diffusion flamelet library at the probability-weighted scalar dissipation χ_{st} from Equation 8.59 (p. 329). For non-adiabatic cases, the flamelet temperature at time t is calculated from

$$T\left(f,t\right)=T^{ad}\left(f,t\right)\xi\left(f,t\right) \tag{8.60}$$

where

$$\xi(f,t)=\frac{\int_V I\rho\overline{T}(\vec{x}|f)\,/\,T^{ad}(f,t)dV}{\int_V I\rho dV} \tag{8.61}$$

Here, the subscript n referring to n^{th} flamelet has been omitted for simplicity, and $\overline{T}\left(\vec{x}\middle|f\right)$ represents the Ansys Fluent steady-state mean cell temperature conditioned on the local cell mixture fraction.

Unsteady flamelet mean species mass fractions in each cell are accumulated over time and can be expressed as:

$$\widetilde{Y}_k=\frac{\sum_n\int_t I_n(t)\left\langle Y_{k,n}\right\rangle dt}{\sum_n\int_t I_n(t)dt} \tag{8.62}$$

where $\widetilde{Y}_k$ denotes the k^{th} species unsteady flamelet mass fraction, and $Y_{k,n}$ is the k^{th} unsteady flamelet mass fraction of n^{th} flamelet and is calculated using a Beta pdf as,

$$\left\langle Y_{k,n}\right\rangle=\int_0^1 Y_{k,n}(f)P(f)df \tag{8.63}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

The probability marker equation (Equation 8.57 (p. 328)) and the flamelet species equation (Equation 8.47 (p. 323)) are advanced together in time until the probability marker has substantially convected and diffused out of the domain. The unsteady flamelet mean species, calculated from Equation 8.61 (p. 329), reach steady-state as the probability marker I_n vanishes.

8.1.6.1.1. Liquid Reactions

Liquid reactors are typically characterized by:

- Near constant density and temperature.
- Relatively slow reactions and species far from chemical equilibrium.
- High Schmidt number (Sc) and hence reduced molecular diffusion.

The Eulerian unsteady laminar flamelet model can be used to model liquid reactions. When the **Liquid Micro-Mixing** model is enabled, Ansys Fluent uses the **volume-weighted-mixing-law** formula to calculate the density.

The effect of high Sc is to decrease mixing at the smallest (micro) scales and increase the mixture fraction variance, which is modeled with the Turbulent Mixer Model [38] (p. 1055). Three transport equations are solved for the inertial-convective ($\overline{f_{ic}^{'2}}$), viscous-convective ($\overline{f_{vc}^{'2}}$), and viscous-diffusive ($\overline{f_{vd}^{'2}}$) subranges of the turbulent scalar spectrum,

$$\frac{\partial}{\partial t}\left(\rho\overline{f_{ic}^{'2}}\right)+\nabla\cdot\left(\rho\vec{v}\overline{f_{ic}^{'2}}\right)=\nabla\cdot\left(\left(\frac{k}{C_p}+\frac{\mu_t}{\sigma_t}\right)\nabla\overline{f_{ic}^{'2}}\right)+C_1\mu_t\cdot\left(\nabla\overline{f}\right)^2-C_2\rho\frac{\varepsilon}{k}\overline{f_{ic}^{'2}} \tag{8.64}$$

$$\frac{\partial}{\partial t}\left(\rho\overline{f_{vc}^{'2}}\right)+\nabla\cdot\left(\rho\vec{v}\overline{f_{vc}^{'2}}\right)=\nabla\cdot\left(\left(\frac{k}{C_p}+\frac{\mu_t}{\sigma_t}\right)\nabla\overline{f_{vc}^{'2}}\right)+C_2\rho\frac{\varepsilon}{k}\overline{f_{ic}^{'2}}-C_3\rho\sqrt{\frac{\varepsilon}{\nu}}\cdot\overline{f_{vc}^{'2}} \tag{8.65}$$

$$\frac{\partial}{\partial t}\left(\rho\overline{f_{vd}^{'2}}\right)+\nabla\cdot\left(\rho\vec{v}\overline{f_{vd}^{'2}}\right)=\nabla\cdot\left(\left(\frac{k}{C_p}+\frac{\mu_t}{\sigma_t}\right)\nabla\overline{f_{vd}^{'2}}\right)+C_3\rho\sqrt{\frac{\varepsilon}{\nu}}\cdot\left(\overline{f_{vc}^{'2}}-\left(C_4+\frac{C_5}{Sc}\right)\overline{f_{vd}^{'2}}\right) \tag{8.66}$$

where k is laminar thermal conductivity of the mixture, C_p is the mixture specific heat, and σ_t is the Prandtl number. The constants C_1 through C_5 have values of 2, 1.86, 0.058, 0.303, and 17050, respectively. The total mixture fraction variance is the sum of $\overline{f_{ic}^{'2}}$, $\overline{f_{vc}^{'2}}$ and $\overline{f_{vd}^{'2}}$.

In Equation 8.66 (p. 330), the cell Schmidt number, Sc, is calculated as $Sc=\mu/\rho D$ where μ is the viscosity, ρ the density, and D the mass diffusivity as defined for the *pdf-mixture* material.

8.1.6.2. The Diesel Unsteady Laminar Flamelet Model

In diesel engines, fuel sprayed into the cylinder evaporates, mixes with the surrounding gases, and then auto-ignites as compression raises the temperature and pressure. The diesel unsteady laminar flamelet model, based on the work of Pitsch et al. and Barths et al. [519] (p. 1083), [45] (p. 1055), models the chemistry by a finite number of one-dimensional laminar flamelets. By reducing the costly chemical kinetic calculation to 1D, substantial savings in run time can be achieved over the laminar-finite-rate, EDC or PDF Transport models.

The flamelet species and energy equations (Equation 8.47 (p. 323) and Equation 8.48 (p. 323)) are solved simultaneously with the flow. The flamelet equations are advanced for a fractional step using

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

properties from the flow, and then the flow is advanced for the same fractional time step using properties from the flamelet.

The initial condition of each flamelet at the time of its introduction into computational domain is a mixed-but-unburnt distribution. For the flamelet fractional time step, the volume-averaged scalar dissipation and pressure, as well as the fuel and oxidizer temperatures, are passed from the flow solver to the flamelet solver. To account for temperature rise during compression, the flamelet energy equation (Equation 8.48 (p. 323)) has an additional term on the right-hand side as

$$\dot{q}=\frac{1}{c_p}\frac{\partial p}{\partial t} \tag{8.67}$$

where c_p is the specific heat and p is the volume-averaged pressure in the cylinder. This rise in flamelet temperature due to compression eventually leads to ignition of the flamelet.

After the flamelet equations have been advanced for the fractional time step, the PDF Table is created as a Non-Adiabatic Steady Flamelet table (see Non-Adiabatic Steady Diffusion Flamelets (p. 327)). Using the properties from this table, the CFD flow field is then advanced for the same fractional time step.

8.1.6.3. Multiple Diesel Unsteady Flamelets

In certain applications where the ignition in different regions of the combustion domain occurs at different times, the chemistry cannot be accurately represented by a single flamelet. Among the examples are split-injections and sprays with high residence times. In these cases, evaporated spray injected at early stage ignites before spray injected at later stage due to the longer residence time. A single flamelet cannot model the local ignition delay for the late spray as the single flamelet represents a burnt state. This deficiency is overcome with the use of multiple flamelets, which are generated in the reacting domain at user-specified times during the simulation. The new flamelet inherits the preceding flamelet boundary temperature calculated at the time of the new flamelet introduction, and the flamelet species field is initialized as mixed-but-unburnt.

The marker probability equations (Equation 8.57 (p. 328)) are solved for $N-1$ flamelets, where N is the total number of unsteady flamelets. The marker probability of the last flamelet is obtained as follows:

$$I_N=\overline{f}-\sum_{n=1}^{N-1}I_n \tag{8.68}$$

where $\overline{f}$ is the mean mixture fraction, and I_n is the marker probability of the n^{th} unsteady flamelet. The scalar dissipation of the n^{th} flamelet is calculated using Equation 8.59 (p. 329).

The properties from the PDF tables $\varphi(t)$ (such as mass fraction, specific heat, and so on) are calculated using the weighted contribution from each flamelet:

$$\langle\varphi(t)\rangle=\frac{\sum_{n=1}^{N-1}I_n(t)\langle\varphi_n(t)\rangle}{\sum_{n=1}^{N-1}I_n(t)} \tag{8.69}$$

where φ_n is the property for n^{th} flamelet.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

The diesel unsteady flamelet approach can model ignition as well as formation of product, intermediate, and pollutant species. The setting of the **Diesel Unsteady Flamelet** model is described in Using the Diesel Unsteady Laminar Flamelet Model in the *Fluent User's Guide*.

8.1.6.4. Multiple Diesel Unsteady Flamelets with Flamelet Reset

As the multiple diesel unsteady flamelets ignite, their species and temperature fields tend toward the same chemical equilibrium state as the scalar dissipation (mixing) decreases. To model multiple engine cycles, Ansys Fluent allows flamelet reset events, which resets multiple flamelets to a single flamelet. Additionally, at the end of a cycle, some burnt gases typically remain trapped and mix later with the fresh charge introduced in the next cycle. The presence of the exhaust gas in the engine chamber can be modeled using either a **Diesel Unsteady Flamelet Reset** or the inert model (via **Inert EGR Reset**).

For information on how to use and set up the **Diesel Unsteady Flamelet Reset** and **Inert EGR Reset** options, see Resetting Diesel Unsteady Flamelets in the *Fluent User's Guide*.

8.1.6.4.1. Resetting the Flamelets

At the end of combustion stroke and just before the inlet valves open, all existing flamelets, except the first one, are deleted, and the probability of the first flamelet is set as:

$$I_1 = \sum_{n=1}^{n=N} I_n$$

where, I_n is the probability of the n^{th} flamelet, and N is the total number of flamelets. The mixture composition (and similarly the temperature) of the reset flamelet is set as:

$$\langle Y_k \rangle_1 = \sum_{n=1}^{N} I_n(t) \langle Y_k \rangle_n$$

where $\langle Y_k \rangle_n$ is the species mass fraction of the k^{th} species from the n^{th} flamelet.

After resetting the flamelets, the computations are performed with this single, typically burnt, flamelet. The flamelet start time of subsequent flamelets should be set to just before the fuel injection. Hence, the injected fuel is modeled in the second and higher flamelets. The process is controlled through dynamic mesh events described in Diesel Unsteady Flamelet Reset in the *Fluent User's Guide*.

8.2. Premixed Combustion

Ansys Fluent has several models to simulate premixed turbulent combustion. For more information about using the premixed turbulent combustion models, see Modeling Premixed Combustion in the User's Guide. Theoretical information about these models is provided in the following sections:

8.2.1. Overview

8.2.2. C-Equation Model Theory

8.2.3. G-Equation Model Theory

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

8.2.4. Turbulent Flame Speed Models
8.2.5. Calculation of Properties

8.2.1. Overview

In premixed combustion, fuel and oxidizer are mixed at the molecular level prior to ignition. Combustion occurs as a flame front propagating into the unburnt reactants. Examples of premixed combustion include aspirated internal combustion engines, lean-premixed gas turbine combustors, and gas-leak explosions.

Premixed combustion is much more difficult to model than non-premixed combustion. The reason for this is that premixed combustion usually occurs as a thin, propagating flame that is stretched and contorted by turbulence. For subsonic flows, the overall rate of propagation of the flame is determined by both the laminar flame speed and the turbulent eddies. The laminar flame speed is determined by the rate that species and heat diffuse upstream into the reactants and burn. To capture the laminar flame speed, the internal flame structure would need to be resolved, as well as the detailed chemical kinetics and molecular diffusion processes. Since practical laminar flame thicknesses are of the order of millimeters or smaller, resolution requirements are usually unaffordable.

The effect of turbulence is to wrinkle and stretch the propagating laminar flame sheet, increasing the sheet area and, in turn, the effective flame speed. The large turbulent eddies tend to wrinkle and corrugate the flame sheet, while the small turbulent eddies, if they are smaller than the laminar flame thickness, may penetrate the flame sheet and modify the laminar flame structure.

Non-premixed combustion, in comparison, can be greatly simplified to a mixing problem (see the mixture fraction approach in Introduction (p. 295)). The essence of premixed combustion modeling lies in capturing the turbulent flame speed, which is influenced by both the laminar flame speed and the turbulence.

In premixed flames, the fuel and oxidizer are intimately mixed before they enter the combustion device. Reaction then takes place in a combustion zone that separates unburnt reactants and burnt combustion products. Partially premixed flames exhibit the properties of both premixed and diffusion flames. They occur when an additional oxidizer or fuel stream enters a premixed system, or when a diffusion flame becomes lifted off the burner so that some premixing takes place prior to combustion.

Premixed and partially premixed flames can be modeled using Ansys Fluent's finite-rate/eddy-dissipation formulation (see Species Transport and Finite-Rate Chemistry (p. 237)). If finite-rate chemical kinetic effects are important, the Laminar Finite-Rate model (see Direct Use of Finite-Rate Kinetics (no TCI) (p. 243)), the EDC model (see The Eddy-Dissipation-Concept (EDC) Model (p. 249)) or the composition PDF transport model (see Composition PDF Transport (p. 276)) can be used. For information about Ansys Fluent's partially premixed combustion model, see Partially Premixed Combustion (p. 343). If the flame is perfectly premixed (all streams entering the combustor have the same equivalence ratio), it is possible to use the premixed combustion model, as described in this section.

For limitations that exist with the premixed combustion model in Ansys Fluent, see Limitations of the Premixed Combustion Model in the *Fluent User's Guide*.

8.2.2. C-Equation Model Theory

A scalar variable representing the progress of reaction from unburnt to burnt is denoted by c. The transport equation for c describes the spatial and temporal evolution of the reaction progress in a

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

turbulent flow field. Ahead of the flame, c is defined as zero in the unburnt reactants, and behind the flame c is unity in the burnt products. Within the flame brush c varies between zero and one. The flame brush propagates upstream at a modeled turbulent flame speed. Fluent offers two models for the turbulent flame speed, namely the Zimont model [732] (p. 1095), [733] (p. 1095), [735] (p. 1095) and the Peters model [511] (p. 1083), as detailed in Peters Flame Speed Model (p. 339).

For more information, see the following section:

8.2.2.1. Propagation of the Flame Front

8.2.2.1. Propagation of the Flame Front

In many industrial premixed systems, combustion takes place in a thin flame sheet. As the flame front moves, combustion of unburnt reactants occurs, converting unburnt premixed reactants to burnt products. The premixed combustion model therefore considers the reacting flow field to be divided into regions of burnt and unburnt species, separated by the flame sheet. Note that the c-equation model assumes that the laminar flame is thin in comparison to the turbulent flame brush, so a value of reaction progress between 0 and 1 implies that the fluctuating flame spends some time at the unburnt state and the remainder at the burnt state; it does not represent an intermediate reaction state between unburnt and burnt.

The flame front propagation is modeled by solving a transport equation for the density-weighted mean reaction progress variable, denoted by $\overline{c}$:

$$\frac{\partial}{\partial t}\left(\rho\overline{c}\right)+\nabla\cdot\left(\rho\vec{v}\overline{c}\right)=\nabla\cdot\left(\left(\frac{k}{C_p}+\frac{\mu_t}{Sc_t}\right)\nabla\overline{c}\right)+\rho S_c \tag{8.70}$$

where

$\overline{c}$ = mean reaction progress variable

Sc_t = turbulent Schmidt number = 0.7

S_c = reaction progress source term (s^{-1})

k = laminar thermal conductivity of the mixture

C_p = mixture specific heat

The progress variable is defined as a normalized sum of the product species mass fractions,

Based on this definition, $\overline{c}=0$ where the mixture is unburnt and $\overline{c}=1$ where the mixture is burnt:

- $\overline{c}=0$: unburnt mixture
- $\overline{c}=1$: burnt mixture

The value of $\overline{c}$ is defined as a boundary condition at all flow inlets. It is usually specified as either 0 (unburnt) or 1 (burnt).

The mean reaction rate in Equation 8.70 (p. 334) is modeled as [733] (p. 1095):

$$\rho S_c=\rho_u U_t\left|\nabla\overline{c}\right| \tag{8.71}$$

where

ρ_u = density of unburnt mixture

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

U_t = turbulent flame speed

Ansys Fluent provides two models for the turbulent flame speed. Many other models for U_t exist [77] (p. 1057) and can be specified using user-defined functions. More information about user-defined functions can be found in the Fluent Customization Manual.

8.2.3. G-Equation Model Theory

The G-equation is a premixed flame-front tracking model. The transport equation governing the unsteady evolution of a propagating flame interface is (derivation can be found in [511] (p. 1083)),

$$\rho\frac{\partial G}{\partial t}+\rho v\cdot\nabla G=\rho U_l|\nabla G|-\rho D\kappa|\nabla G| \tag{8.72}$$

where ρ is the fluid density, v is the fluid velocity vector, U_l is the laminar flame speed (interface normal propagation), D is the diffusivity and κ is the flame curvature. For turbulent flames, Equation 8.72 (p. 335) can be Favre Reynolds-averaged or spatially-filtered to provide transport equations for the flame mean position,

$$\tilde{\rho}\frac{\partial \overline{G}}{\partial t}+\tilde{\rho}\tilde{v}\cdot\nabla\overline{G}=\rho_u U_t|\nabla\overline{G}|-\rho_u D_t'\tilde{\kappa}|\nabla\overline{G}| \tag{8.73}$$

and the variance of the flame position,

$$\tilde{\rho}\frac{\partial \overline{G}''^2}{\partial t}+\tilde{\rho}\tilde{v}\cdot\nabla\overline{G}''^2=\nabla_|\cdot(\rho_u D_t\nabla_|\overline{G}''^2)+2\rho_u D_t(\nabla\overline{G})^2-c_s\rho_u\frac{u'}{l_t}\overline{G}''^2 \tag{8.74}$$

where

U_t is the turbulent flame speed

D_t' is a diffusivity term defined as $D_t'=\sqrt{\frac{c_\mu c_s}{2Sc_t}}\,l_t u'$

u' is the turbulent velocity scale

c_μ is a modeling constant taken from the $k-\varepsilon$ turbulence model (default=0.09)

c_s is a modeling constant = 2.0

Sc_t is the turbulent Schmidt number

l_t is the turbulent length scale

κ is the flame curvature, defined as $\kappa=\nabla\cdot n$, where $n=-\frac{\nabla\overline{G}}{|\nabla\overline{G}|}$

In Equation 8.74 (p. 335), $\nabla_|$ indicates that the diffusion term is only applied parallel to the flame front [511] (p. 1083), the normal component being accounted for in the turbulent burning velocity. In practice this makes only a minor difference to results and can lead to convergence problems, and is therefore disabled by default.

Ansys Fluent also offers the option of using an algebraic expression to calculate the flame position variance, $\overline{G}''^2$, instead of Equation 8.74 (p. 335):

$$\overline{G}''^2=\frac{l_t}{u'}\frac{\mu_{eff}}{\tilde{\rho}Sc_t} \tag{8.75}$$

where μ_{eff} is the effective turbulent viscosity.

For more information, see the following section:

8.2.3.1. Numerical Solution of the G-equation

8.2.3.1. Numerical Solution of the G-equation

Equation 8.72 (p. 335) and Equation 8.73 (p. 335) do not contain a diffusion term and sharp interfaces remain sharp at all times. Special numerical techniques such as the Volume-of-Fluid (VOF) and Marker-and-Cell (MAC) have been devised to solve such equations. In Ansys Fluent, the mean G-equation Equation 8.73 (p. 335) is solved using a Level-set method. Here, $\overline{G}$ represents the signed mean distance to the flame front, and hence the flame front is the $\overline{G}=0$ isosurface. Since $\overline{G}$ is the distance to the flame front, $\left|\nabla\overline{G}\right|$ is constrained to be unity everywhere in the flow field. The standard Ansys Fluent transport equation machinery is used to solve for $\overline{G}$ (Equation 8.73 (p. 335)) over a time step. However, at the end of the time step, the $\overline{G}$ field is typically is not exactly equal to the mean flame distance (and $\left|\nabla\overline{G}\right|$ is not identically equal to 1), and this condition is enforced by a process called re-initialization. In Ansys Fluent this is done by constructing a faceted representation of the flame front from the $\overline{G}$ field. Then, in every cell, $\overline{G}$ is set to the geometric distance to the nearest flame front facet. $\overline{G}$ is positive in the burnt region downstream of the flame front and negative in the unburnt region upstream of the flame front.

Given the mean flame front position, the mean progress variable is calculated according to a Gaussian distribution depending on proximity to the flame front and the G-equation variance:

$$\overline{c}=\int_{G=0}^{\infty}\frac{1}{\sqrt{2\pi\overline{G}''^{2}}}\exp\left(-\frac{\left(G-\overline{G}\right)^{2}}{2\overline{G}''^{2}}\right)dG \tag{8.76}$$

Mean properties, such as the mean density, temperature and species mass fractions, are calculated from the mean reaction progress variable, as in the C-equation model.

8.2.4. Turbulent Flame Speed Models

The key to the premixed combustion model is the prediction of U_t, the turbulent flame speed normal to the mean surface of the flame. The turbulent flame speed is influenced by the following:

- laminar flame speed, which is, in turn, determined by the fuel concentration, temperature, and molecular diffusion properties, as well as the detailed chemical kinetics
- flame front wrinkling and stretching by large eddies, and flame thickening by small eddies

Ansys Fluent has two turbulent flame speed models, namely the Zimont turbulent flame speed closure model and the Peters flame speed model.

For more information, see the following sections:

8.2.4.1. Zimont Turbulent Flame Speed Closure Model

8.2.4.2. Peters Flame Speed Model

8.2.4.1. Zimont Turbulent Flame Speed Closure Model

In Ansys Fluent, the Zimont turbulent flame speed closure is computed using a model for wrinkled and thickened flame fronts [733] (p. 1095):

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$U_t = A\left(u'\right)^{3/4} U_l^{1/2} \alpha^{-1/4} \ell_t^{1/4} \tag{8.77}$$

$$U_t = MAX\left(U_l, Au'\left(\frac{\tau_t}{\tau_c}\right)^{1/4}\right) \tag{8.78}$$

where

A = model constant

u' = RMS (root-mean-square) velocity (m/s)

U_l = laminar flame speed (m/s)

$\alpha = k / \rho c_p$ = unburnt thermal diffusivity (m^2/s)

ℓ_t = turbulence length scale (m)

$\tau_t = \ell_t / u'$ = turbulence time scale (s)

$\tau_c = \alpha / U_l^2$ = chemical time scale (s)

The turbulence length scale, ℓ_t, is computed from

$$\ell_t = C_D \frac{\left(u'\right)^3}{\varepsilon} \tag{8.79}$$

where ε is the turbulence dissipation rate.

The model is based on the assumption of equilibrium small-scale turbulence inside the laminar flame, resulting in a turbulent flame speed expression that is purely in terms of the large-scale turbulent parameters. The default value of 0.52 for A is recommended [733] (p. 1095), and is suitable for most premixed flames. The default value of 0.37 for C_D should also be suitable for most premixed flames.

The model is strictly applicable when the smallest turbulent eddies in the flow (the Kolmogorov scales) are smaller than the flame thickness, and penetrate into the flame zone. This is called the thin reaction zone combustion region, and can be quantified by Karlovitz numbers, Ka, greater than unity. Ka is defined as

$$Ka = \frac{t_l}{t_\eta} = \frac{v_\eta^2}{U_l^2} \tag{8.80}$$

where

t_l = characteristic flame time scale

t_η = smallest (Kolmogorov) turbulence time scale

$v_\eta = \left(\nu\varepsilon\right)^{1/4}$ = Kolmogorov velocity

ν = kinematic viscosity

8.2.4.1.1. Zimont Turbulent Flame Speed Closure for LES

For simulations that use the LES turbulence model, the Reynolds-averaged quantities in the turbulent flame speed expression (Equation 8.77 (p. 337)) are replaced by their equivalent subgrid quantities. In particular, the large eddy length scale ℓ_t is modeled as

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$\ell_t = C_s \Delta \tag{8.81}$$

where C_s is the Smagorinsky constant and Δ is the cell characteristic length.

The RMS velocity in Equation 8.77 (p. 337) is replaced by the subgrid velocity fluctuation, calculated as

$$u' = \ell_t \tau_{sgs}^{-1} \tag{8.82}$$

where τ_{sgs}^{-1} is the subgrid scale mixing rate (inverse of the subgrid scale time scale), given in Equation 7.41 (p. 248).

8.2.4.1.2. Flame Stretch Effect

Since industrial low-emission combustors often operate near lean blow-off, flame stretching will have a significant effect on the mean turbulent heat release intensity. To take this flame stretching into account, the source term for the progress variable (ρS_c in Equation 8.70 (p. 334)) is multiplied by a stretch factor, G [735] (p. 1095). This stretch factor represents the probability that the stretching will not quench the flame; if there is no stretching ($G=1$), the probability that the flame will be unquenched is 100%.

The stretch factor, G, is obtained by integrating the log-normal distribution of the turbulence dissipation rate, ε:

$$G = \frac{1}{2} erfc\left\{ -\sqrt{\frac{1}{2\sigma}} \left[\ln\left(\frac{\varepsilon_{cr}}{\varepsilon}\right) + \frac{\sigma}{2} \right] \right\} \tag{8.83}$$

where $erfc$ is the complementary error function, and σ and ε_{cr} are defined below. For LES, ε is calculated by:

$$\varepsilon = \frac{u'^2}{\tau} \tag{8.84}$$

where τ is the time scale and u' is the is subgrid scale (SGS) velocity fluctuations, which depends on the SGS closure. For details on how u' is calculated, refer to Subgrid-Scale Models (p. 120).

σ is the standard deviation of the distribution of ε:

$$\sigma = \mu_{str} \ln\left(\frac{L}{\eta}\right) \tag{8.85}$$

where μ_{str} is the stretch factor coefficient for dissipation pulsation, L is the turbulent integral length scale, and η is the Kolmogorov micro-scale. The default value of 0.26 for μ_{str} (measured in turbulent non-reacting flows) is recommended by [733] (p. 1095), and is suitable for most pre-mixed flames.

ε_{cr} is the turbulence dissipation rate at the critical rate of strain [733] (p. 1095):

$$\varepsilon_{cr} = 15 \nu g_{cr}^2 \tag{8.86}$$

By default, g_{cr} is set to a very high value (1×10^8) so no flame stretching occurs. To include flame stretching effects, the critical rate of strain g_{cr} should be adjusted based on experimental data for the burner. Numerical models can suggest a range of physically plausible values [733] (p. 1095), or an appropriate value can be determined from experimental data. A reasonable model for the critical rate of strain g_{cr} is

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$g_{cr}=\frac{BU_l^2}{\alpha} \tag{8.87}$$

where B is a constant (typically 0.5) and α is the unburnt thermal diffusivity. Equation 8.87 (p. 339) can be implemented in Ansys Fluent using a property user-defined function. More information about user-defined functions can be found in the Fluent Customization Manual.

8.2.4.1.3. Wall Damping

High turbulent kinetic energy levels at the walls in some problems can cause an unphysical acceleration of the flame along the wall. In reality, radical quenching close to walls decreases reaction rates and therefore the flame speed, but is not included in the model. To approximate this effect, Ansys Fluent includes a constant multiplier for the turbulent flame speed, α_w,which modifies the flame speed in the vicinity of wall boundaries:

$$U_t=\alpha_w A u' \left(\frac{\tau_t}{\tau_c}\right)^{1/4} \tag{8.88}$$

The default for this constant is 1 which does not change the flame speed. Values of α_w larger than 1 increase the flame speed, while values less than 1 decrease the flame speed in the cells next to the wall boundary.

Ansys Fluent will solve the transport equation for the reaction progress variable $\overline{c}$ (Equation 8.70 (p. 334)), computing the source term, ρS_c, based on the theory outlined above:

$$\begin{aligned} \rho S_c &= AG\rho_u u'^{3/4}\left[U_l\right]^{1/2}\left[\alpha\right]^{-1/4}\ell_t^{1/4}\left|\nabla \overline{c}\right| \\ &= AG\rho_u u'\left[\frac{\tau_t}{\tau_c}\right]^{1/4}\left|\nabla \overline{c}\right| \end{aligned} \tag{8.89}$$

8.2.4.2. Peters Flame Speed Model

The Peters model [511] (p. 1083) for turbulent flame speed is used in the form proposed by Ewald [167] (p. 1062):

$$U_t = U_l\left(1+\sigma_t\right) \tag{8.90}$$

where

$$\sigma_t=-\frac{a_4 b_3^2}{4b_1}\frac{l_f}{\delta}l_*^q+\left[\left(\frac{a_4 b_3^2}{4b_1}\frac{l_f}{\delta}l_*^q\right)^2+\frac{c_s b_3^2}{2}\frac{u' l_f^2}{U_l \delta l_t}\right]^{1/2} \tag{8.91}$$

The term l_*^q is Ewald's corrector and may be disabled by you, in which case $l_*^q=1$ and the formulation reduces to that of Peters flame speed model.

In Equation 8.91 (p. 339):

U_l is the laminar flame speed

$\delta=\left(\frac{\lambda/c_p}{\rho U_l}\right)_{unburned}$ is the laminar flame thickness

u' is the turbulent velocity scale

l_f is the flame brush thickness, given below

$l_*^q = \frac{l_f}{l_{alg}}$

$l_{alg} = \sqrt{\frac{2C_\mu}{C_s Sc_t}}\, l_t$ is an algebraic flame brush thickness

$a_4 = \sqrt{\frac{3C_\mu C_s}{Sc_t}}$

b_3 is a constant with a default value of 1.0

b_1 is a constant with a default value of 2.0

q is a constant with a default value of 0.66

c_μ is a modeling constant taken from the $k-\varepsilon$ turbulence model with a default value of 0.09

C_s is a constant with a default value of 2.0

Sc_t is the turbulent Schmidt number with a default value of 0.7

If the Blint modifier is being used

$$\delta = \left(\frac{\lambda / C_p}{\rho U_l} \right)_{unburned} B_1 \left(\frac{\rho_u}{\rho_b} \right)^{B_2} \tag{8.92}$$

where

B_1 = 2.0

B_2 = 0.7

ρ_u is the unburned density

ρ_b is the burned density

The flame brush thickness is calculated differently for the C-equation and G-equation models.

For the G equation model [511] (p. 1083):

$$l_f = \frac{\sqrt{\widetilde{G}''^2}}{|\nabla \widetilde{G}|} \tag{8.93}$$

For the C-equation model an algebraic form of this is used giving:

$$l_f = \sqrt{\frac{l_t}{u'} \frac{\mu_{eff}}{\widetilde{\rho} Sc_t}} \tag{8.94}$$

Note that for the C-equation model the Ewald corrector has no affect.

To reduce the flame speed along walls, Ansys Fluent includes a constant multiplier for the turbulent flame speed, α_w, which modifies the flame speed in the vicinity of wall boundaries by multiplying the U_t expression in Equation 8.90 (p. 339) by a constant between 0 and 1.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

8.2.4.2.1. Peters Flame Speed Model for LES

For LES the Peters model [511] (p. 1083) for turbulent flame speed must be modified to use the subgrid quantities. Here we use the form derived by Pitsch [518] (p. 1083).

$$S_t = S_l.MAX\left(1.0,\sigma_t\right) \tag{8.95}$$

$$\sigma_t = -\frac{b_3^2}{2b_1 Sc_t}\frac{1}{\delta}\frac{\nu_t}{u'} + \left[\left(\frac{b_3^2}{2b_1 Sc_t}\frac{1}{\delta}\frac{\nu_t}{u'}\right)^2 + \frac{b_3^2}{Sc_t}\frac{\nu_t}{D}\right]^{1/2} \tag{8.96}$$

where $D = \left(\frac{\lambda/c_p}{\rho}\right)_{unburned}$ is the thermal diffusivity.

The turbulent velocity scale u' is the subgrid quantity as given in Equation 8.82 (p. 338).

8.2.5. Calculation of Properties

The C-equation and G-equation turbulent premixed combustion models require properties for temperature, density, unburnt density, unburnt thermal diffusivity and laminar flame speed, which are modeled as described in the sections that follow.

For more information, see the following sections:

8.2.5.1. Calculation of Temperature

8.2.5.2. Calculation of Density

8.2.5.3. Laminar Flame Speed

8.2.5.4. Unburnt Density and Thermal Diffusivity

8.2.5.1. Calculation of Temperature

The calculation method for temperature will depend on whether the model is adiabatic or non-adiabatic.

8.2.5.1.1. Adiabatic Temperature Calculation

For the adiabatic premixed combustion model, the temperature is assumed to be a linear function of reaction progress between the lowest temperature of the unburnt mixture, T_u, and the highest adiabatic burnt temperature T_{ad}:

$$T = \left(1-\overline{c}\right)T_u + \overline{c}T_{ad} \tag{8.97}$$

8.2.5.1.2. Non-Adiabatic Temperature Calculation

For the non-adiabatic premixed combustion model, Ansys Fluent solves an energy transport equation in order to account for any heat losses or gains within the system. The energy equation in terms of sensible enthalpy, h, for the fully premixed fuel (see Equation 5.2 (p. 172)) is as follows:

$$\frac{\partial}{\partial t}\left(\rho h\right) + \nabla\cdot\left(\rho\vec{v}h\right) = \nabla\cdot\left(\frac{k+k_t}{c_p}\nabla h\right) + S_{h,chem} + S_{h,rad} \tag{8.98}$$

$S_{h,rad}$ represents the heat losses due to radiation and $S_{h,chem}$ represents the heat gains due to chemical reaction:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$S_{h,chem}=\rho S_c H_{comb} Y_{fuel} \tag{8.99}$$

where

S_c = normalized average rate of product formation (s^{-1})

H_{comb} = heat of combustion for burning 1 kg of fuel (J/kg)

Y_{fuel} = fuel mass fraction of unburnt mixture

8.2.5.2. Calculation of Density

Ansys Fluent calculates the premixed density using the ideal gas law. For the adiabatic model, pressure variations are neglected and the mean molecular weight is assumed to be constant. The burnt gas density is then calculated from the following relation:

$$\rho_b T_b=\rho_u T_u \tag{8.100}$$

where the subscript u refers to the unburnt cold mixture, and the subscript b refers to the burnt hot mixture. The required inputs are the unburnt density (ρ_u), the unburnt temperature (T_u), and the burnt adiabatic flame temperature (T_b).

For the non-adiabatic model, you can choose to either include or exclude pressure variations in the ideal gas equation of state. If you choose to ignore pressure fluctuations, Ansys Fluent calculates the density from

$$\rho T=\rho_u T_u \tag{8.101}$$

where T is computed from the energy transport equation, Equation 8.98 (p. 341). The required inputs are the unburnt density (ρ_u) and the unburnt temperature (T_u). Note that, from the incompressible ideal gas equation, the expression $\rho_u R T_u / p_{op}$ may be considered to be the effective molecular weight of the gas, where R is the gas constant and p_{op} is the operating pressure.

If you want to include pressure fluctuations for a compressible gas, you will need to specify the effective molecular weight of the gas, and the density will be calculated from the ideal gas equation of state.

8.2.5.3. Laminar Flame Speed

The laminar flame speed (U_l in Equation 8.71 (p. 334)) can be specified as constant, or as a user-defined function. A third option appears for non-adiabatic premixed and partially-premixed flames and is based on the correlation proposed by Metghalchi and Keck [434] (p. 1078),

$$U_l=U_{l,ref}\left(\frac{T_u}{T_{u,ref}}\right)^{\gamma}\left(\frac{p_u}{p_{u,ref}}\right)^{\beta} \tag{8.102}$$

In Equation 8.102 (p. 342), T_u and p_u are the unburnt reactant temperature and pressure ahead of the flame, $T_{u,ref}=298\,\mathrm{K}$ and $p_{u,ref}=1\,\mathrm{atm}$.

The reference laminar flame speed, $U_{l,ref}$, is calculated from

$$U_{l,ref}=C_1+C_2\left(\varphi-C_3\right)^2 \tag{8.103}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

where φ is the equivalence ratio ahead of the flame front, and C_1, C_2 and C_3 are fuel-specific constants. The exponents γ and β are calculated from,

$$\gamma = 2.18 - 0.8(\varphi - 1)$$
$$\beta = -0.16 + 0.22(\varphi - 1) \tag{8.104}$$

The Metghalchi-Keck laminar flame speeds are available for fuel-air mixtures of methane, methanol, propane, iso-octane and indolene fuels.

8.2.5.4. Unburnt Density and Thermal Diffusivity

The unburnt density (ρ_u in Equation 8.71 (p. 334)) and unburnt thermal diffusivity (α in Equation 8.77 (p. 337) and Equation 8.78 (p. 337)) are specified constants that are set in the **Materials** dialog box. However, for compressible cases, such as in-cylinder combustion, these can change significantly in time and/or space. When the ideal gas model is selected for density, the unburnt density and thermal diffusivity are calculated by evaluating the local cell at the unburnt state c=0.

8.3. Partially Premixed Combustion

Ansys Fluent provides a partially premixed combustion model that is based on the non-premixed combustion model described in Non-Premixed Combustion (p. 295) and the premixed combustion model described in Premixed Combustion (p. 332). For information about using the partially premixed combustion model, see Modeling Partially Premixed Combustion in the User's Guide. Information about the theory behind the partially premixed combustion model is presented in the following sections:

8.3.1. Overview

8.3.2. Partially Premixed Combustion Theory

8.3.1. Overview

Partially premixed combustion systems are premixed flames with non-uniform fuel-oxidizer mixtures (equivalence ratios). Such flames include premixed jets discharging into a quiescent atmosphere, lean premixed combustors with diffusion pilot flames and/or cooling air jets, and imperfectly premixed inlets. Ansys Fluent has three types of partially premixed models, namely Chemical Equilibrium, Steady Diffusion Flamelet, and Flamelet Generated Manifold.

The Chemical Equilibrium and Steady Diffusion Flamelet partially-premixed models assume that the premixed flame front is infinitely thin, with unburnt reactants ahead and burnt products behind the flame front. The composition of the burnt products can be modeled assuming chemical equilibrium or with steady laminar diffusion flamelets. The flame-brush is indicated by a value of mean reaction progress $\overline{c}$ between 0 and 1. Note that at a point within the turbulent premixed flame brush, $0 < \overline{c} < 1$, the fluctuating thin flame spends some time at the unburnt state ($c=0$) and the remaining time at the burnt state ($c=1$), with a mean reaction progress between zero and one. It should not be interpreted that the instantaneous premixed flame reaction progress is intermediate between the unburnt and burnt state.

The Flamelet Generated Manifold (FGM) model [663] (p. 1091) assumes that the thermochemical states in a turbulent flame are similar to the states in a laminar flame, and parameterize these by mixture fraction and reaction progress. Within the laminar flame, reaction progress increases from $c=0$ in the unburnt reactants to $c=1$ in the burnt products, over a nonzero flame thickness. A point within the

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

turbulent flame brush with $0< \overline{c}< 1$ has contributions from both fluctuating flame fronts, as well as intermediate reaction progress. Ansys Fluent has the option to model the FGM with either premixed or diffusion laminar flames.

For limitations that apply to the partially premixed combustion model, see Limitations in the *Fluent User's Guide*.

8.3.2. Partially Premixed Combustion Theory

The partially premixed model solves a transport equation for the mean reaction progress variable, $\overline{c}$, or the mean flame position, $\overline{G}$ (to determine the position of the flame front), as well as the mean mixture fraction, $\overline{f}$ and the mixture fraction variance, $\overline{f'^2}$. The Flamelet Generated Manifold model has an option to solve a transport equation for the reaction progress variable variance, $\overline{c'^2}$, or to use an algebraic expression. Ahead of the flame ($\overline{c}=0$), the fuel and oxidizer are mixed but unburnt and behind the flame ($\overline{c}=1$), the mixture is burnt.

For more information, see the following sections:

8.3.2.1. Chemical Equilibrium and Steady Diffusion Flamelet Models
8.3.2.2. Flamelet Generated Manifold (FGM) Model
8.3.2.3. FGM Turbulent Closure
8.3.2.4. Calculation of Mixture Properties
8.3.2.5. Calculation of Unburnt Properties
8.3.2.6. Laminar Flame Speed
8.3.2.7. Strained Laminar Flame Speed
8.3.2.8. Generating PDF Lookup Tables Through Automated Grid Refinement

8.3.2.1. Chemical Equilibrium and Steady Diffusion Flamelet Models

Density weighted mean scalars (such as species fractions and temperature), denoted by $\overline{\phi}$, are calculated from the probability density function (PDF) of f and c as

$$\overline{\phi}=\int_0^1\int_0^1 \phi(f,c)p(f,c)dfdc \tag{8.105}$$

Under the assumption of thin flames, so that only unburnt reactants and burnt products exist, the mean scalars are determined from

$$\overline{\phi}=\overline{c}\int_0^1 \phi_b(f)p(f)df+(1-\overline{c})\int_0^1 \phi_u(f)p(f)df \tag{8.106}$$

where the subscripts b and u denote burnt and unburnt, respectively.

The burnt scalars, ϕ_b, are functions of the mixture fraction and are calculated by mixing a mass f of fuel with a mass $(1-f)$ of oxidizer and allowing the mixture to equilibrate. When non-adiabatic mixtures and/or diffusion laminar flamelets are considered, ϕ_b is also a function of enthalpy and/or

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

strain, but this does not alter the basic formulation. The unburnt scalars, ϕ_u, are calculated similarly by mixing a mass f of fuel with a mass $\left(1-f\right)$ of oxidizer, but the mixture is not reacted.

Just as in the non-premixed model, the chemistry calculations and PDF integrations for the burnt mixture are performed in Ansys Fluent, and look-up tables are constructed.

It is important to understand that in the limit of perfectly premixed combustion, the equivalence ratio and hence mixture fraction is constant. Hence, the mixture fraction variance and its scalar dissipation are zero. If you are using laminar diffusion flamelets, the flamelet at the lowest strain will always be interpolated, and if you have **Include Equilibrium Flamelet** enabled, the Ansys Fluent solution will be identical to a calculation with a chemical equilibrium PDF table.

8.3.2.2. Flamelet Generated Manifold (FGM) Model

The Laminar Flamelet model (see The Diffusion Flamelet Models Theory (p. 319)) postulates that a turbulent flame is an ensemble of laminar flames that have an internal structure not significantly altered by the turbulence. These laminar flamelets are embedded in the turbulent flame brush using statistical averaging. The Flamelet Generated Manifold (FGM) [663] (p. 1091) model assumes that the scalar evolution (that is the realized trajectories on the thermochemical manifold) in a turbulent flame can be approximated by the scalar evolution in a laminar flame. Both Laminar Flamelet and FGM parameterize all species and temperature by a few variables, such as mixture-fraction, scalar-dissipation and/or reaction-progress, and solve transport equations for these parameters in a 3D CFD simulation.

Note that the FGM model is fundamentally different from the Laminar Flamelet model. For instance, since Laminar Flamelets are parameterized by strain, the thermochemistry always tends to chemical equilibrium as the strain rate decays towards the outlet of the combustor. In contrast, the FGM model is parameterized by reaction progress and the flame can be fully quenched, for example, by adding dilution air. No assumption of thin and intact flamelets is made by FGM, and the model can theoretically be applied to the stirred reactor limit, as well as to ignition and extinction modeling.

Any type of laminar flame can be used to parameterize an FGM. Ansys Fluent can either import an FGM calculated in a third-party flamelet code and written in Standard file format (see Standard Files for Flamelet Generated Manifold Modeling in the User's Guide), or calculate an FGM from 1D steady premixed flamelets or 1D diffusion flamelets.

In general, premixed FGMs should be used for turbulent partially-premixed flames that are predominantly premixed. Similarly, diffusion FGMs should be used for turbulent partially-premixed flames that are predominantly non-premixed. For further instructions on how to use the FGM model, see Flamelet Generated Manifold in the *Fluent User's Guide*.

8.3.2.2.1. Premixed FGMs in Reaction Progress Variable Space

While the only possible configuration for a diffusion flamelet in 1D is opposed flow, 1D steady premixed flamelets can have several configurations. These include unstrained adiabatic freely-propagating, unstrained non-adiabatic burner-stabilized, as well as a strained opposed flow premixed flames.

1D premixed flamelets can be solved in physical space (for example, [543] (p. 1084), [97] (p. 1058)), then transformed to reaction-progress space in the Ansys Fluent premixed flamelet file format, and imported into Ansys Fluent. Alternatively 1D premixed flamelets can be generated in Ansys

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Fluent, which solves the flamelets in reaction-progress space. The reaction progress variable is defined as a normalized sum of the product species mass fractions:

$$c=\frac{\sum_k \alpha_k\left(Y_k-Y_k^u\right)}{\sum_k \alpha_k\left(Y_k^{eq}-Y_k^u\right)}=\frac{Y_c}{Y_c^{eq}} \tag{8.107}$$

where

superscript u denotes the unburnt reactant at the flame inlet

Y_k denotes the k^{th} species mass fraction

superscript eq denotes chemical equilibrium at the flame outlet

α_k are constants that are typically zero for reactants and unity for a few product species

The coefficients α_k should be prescribed so that the reaction progress c increases monotonically through the flame. By default, $\alpha_k=0$ for all species other than $\alpha_{CO_2}=\alpha_{CO}=1$, which are selected for hydrocarbon combustion. In the case where the element C is missing from the chemical mechanism, such as H_2 combustion, Ansys Fluent uses $\alpha_k=0$ for all species other than $\alpha_{HO_2}=10$ and $\alpha_{H_2O}=1$, by default. For hydrogen blended fuels, the coefficients α_k are defined based on the mass fraction of hydrogen in the fuel as:

$$\alpha_{CO}=\alpha_{CO_2}=1-Y_{H_2,fuel}$$
$$\alpha_{HO_2}=10Y_{H_2,fuel}$$
$$\alpha_{H_2O}=Y_{H_2,fuel}$$

where $Y_{H_2,fuel}$ is the mass fraction of the hydrogen in the fuel composition. The coefficients α_k can be specified in the user interface when generating FGM.

The 1D adiabatic premixed flame equations can be transformed from physical-space to reaction-progress space [474] (p. 1080), [385] (p. 1075). Neglecting differential-diffusion, these equations are

$$\rho\frac{\partial Y_k}{\partial t}+\frac{\partial Y_k}{\partial c}\dot{\omega}_c=\rho\chi_c\frac{\partial^2 Y_k}{\partial c^2}+\dot{\omega}_k \tag{8.108}$$

$$\rho\frac{\partial T}{\partial t}+\frac{\partial T}{\partial c}\dot{\omega}_c=\rho\chi_c\frac{\partial^2 T}{\partial c^2}-\frac{1}{c_p}\sum_k h_k\dot{\omega}_k+\frac{\rho\chi_c}{c_p}\left(\frac{\partial c_p}{\partial c}+\sum_k c_{p,k}\frac{\partial Y_k}{\partial c}\right)\frac{\partial T}{\partial c} \tag{8.109}$$

where Y_k is the k^{th} species mass fraction, T is the temperature, ρ is the fluid density, t is time, $\dot{\omega}_k$ is the k^{th} species mass reaction rate, h_k is the total enthalpy, and $c_{p,k}$ is the k^{th} species specific heat at constant pressure.

The scalar-dissipation rate, χ_c in Equation 8.108 (p. 346) and Equation 8.109 (p. 346) is defined as

$$\chi_c=\frac{\lambda}{\rho c_p}\left|\nabla c\right|^2 \tag{8.110}$$

where λ is the thermal conductivity. Note that χ_c varies with c and is an input to the equation set. If $\chi_c(c)$ is taken from a 1D physical-space, adiabatic, equi-diffusivity flamelet calculation, either freely-propagating (unstrained) or opposed-flow (strained), the c-space species and tem-

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

perature distributions would be identical to the physical-space solution. However, $\chi_c(c)$ is generally not known and is modeled in Ansys Fluent as

$$\chi_c(c)=\chi_{\max}\exp\left(-2\left(erfc^{-1}(2c)\right)^2\right) \tag{8.111}$$

where $\chi_{\max}$ is a user-specified maximum scalar dissipation within the premixed flamelet, and $erfc^{-1}$ is the inverse complementary error function.

A 1D premixed flamelet is calculated at a single equivalence ratio, which can be directly related to a corresponding mixture fraction. For partially-premixed combustion, premixed laminar flamelets must be generated over a range of mixture fractions. Premixed flamelets at different mixture fractions have different maximum scalar dissipations, $\chi_{\max}$. In Ansys Fluent, the scalar dissipation $\chi_c(f,c)$ at any mixture fraction f is modeled as

$$\chi_c(f,c)=\chi_{\max}^{sto}\exp\left(-2\left(erfc^{-1}\left(\frac{f}{f_{sto}}\right)\right)^2\right)\exp\left(-2\left(erfc^{-1}(2c)\right)^2\right) \tag{8.112}$$

where sto indicates stoichiometric mixture fraction.

Hence, the only model input to the premixed flamelet generator in Ansys Fluent is the scalar dissipation at stoichiometric mixture fraction, $\chi_{\max}^{sto}$. The default value is $\chi_{\max}^{sto}=1000$ 1/s, which reasonably matches solutions of unstrained (freely propagating) physical-space flamelets for rich, lean, and stoichiometric hydrocarbon and hydrogen flames at standard temperature and pressure.

The following are important points to consider about premixed flamelet generation in Ansys Fluent in general and when specifying $\chi_{\max}^{sto}$ in particular:

- It is common in the FGM approach to use unstrained, freely-propagating premixed flamelets. However, for highly turbulent flames where the instantaneous premixed flame front is stretched and distorted by the turbulence, a strained premixed flamelet may be a better representation of the manifold. The Ansys Fluent FGM model allows a premixed flamelet generated manifold at a single, representative strain rate, $\chi_{\max}^{sto}$.

- Calculating flamelets in physical-space can be compute-intensive and difficult to converge over the entire mixture fraction range, especially with large kinetic mechanisms at the flammability limits. The Ansys Fluent solution in reaction-progress space is substantially faster and more robust than a corresponding physical-space solution. However, if physical-space solutions (for example, from [543] (p. 1084), [97] (p. 1058)) are preferred, they can be generated with an external code, transformed to reaction-progress space, and imported in Ansys Fluent's Standard Flamelet file format.

- A more appropriate value of $\chi_{\max}^{sto}$ than the default (1000/s) for the specific fuel and operating conditions of your combustor can be determined from a physical-space premixed flamelet solution at stoichiometric mixture fraction. This premixed flamelet solution can be unstrained or strained. $\chi_c(c)$ through the flamelet is then calculated from Equation 8.110 (p. 346) and $\chi_{\max}^{sto}$ is the maximum value of $\chi_c(c)$.

- As $\chi_{\max}^{sto}$ is increased, the solution of temperature and species fractions as a function of c tends to the thin-flamelet chemical equilibrium solution, which is linear between unburnt at $c=0$ and chemical equilibrium at $c=1$. The minimum specified $\chi_{\max}^{sto}$ should be that in a freely-propagating flame. For smaller values of $\chi_{\max}^{sto}$, the Ansys Fluent premixed flamelet

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

generator will have difficulties converging. When Ansys Fluent fails to converge steady premixed flamelets at rich mixture fractions, equilibrium thin-flamelet solutions are used.

- The Ansys Fluent partially-premixed model can be used to model partially-premixed flames ranging from the non-premixed to the perfectly-premixed limit. In the non-premixed limit, reaction progress is unity everywhere in the domain and all premixed streams are fully burnt. In the premixed limit, the mixture fraction is constant in the domain. When creating a partially-premixed PDF table, the interface requires specification of the fuel and oxidizer compositions and temperatures. For partially-premixed flames, the fuel composition can be modeled as pure fuel, in which case any premixed inlet would be set to the corresponding mixture fraction, which is less than one. Alternatively, the fuel composition can be specified in the interface as a premixed inlet composition, containing both fuel and oxidizer components. In this latter case, the mixture fraction at premixed inlets would be set to one.

- You can use the automated grid refinement (AGR) when solving premixed flamelets. In the context of the FGM model, the grid in the reaction-progress space is refined. For information on the AGR, see Steady Diffusion Flamelet Automated Grid Refinement (p. 326).

8.3.2.2.2. Premixed FGMs in Physical Space

The premixed FGM flamelets solved in reaction progress space using Equation 8.108 (p. 346) and Equation 8.109 (p. 346) require a closure of the scalar dissipation term. Ansys Fluent calculates the scalar dissipation in the progress variable using Equation 8.112 (p. 347), which is controlled by the following two parameters:

- Maximum value of the scalar dissipation for each mixture fraction
- Distribution of the scalar dissipation with respect to the reaction progress

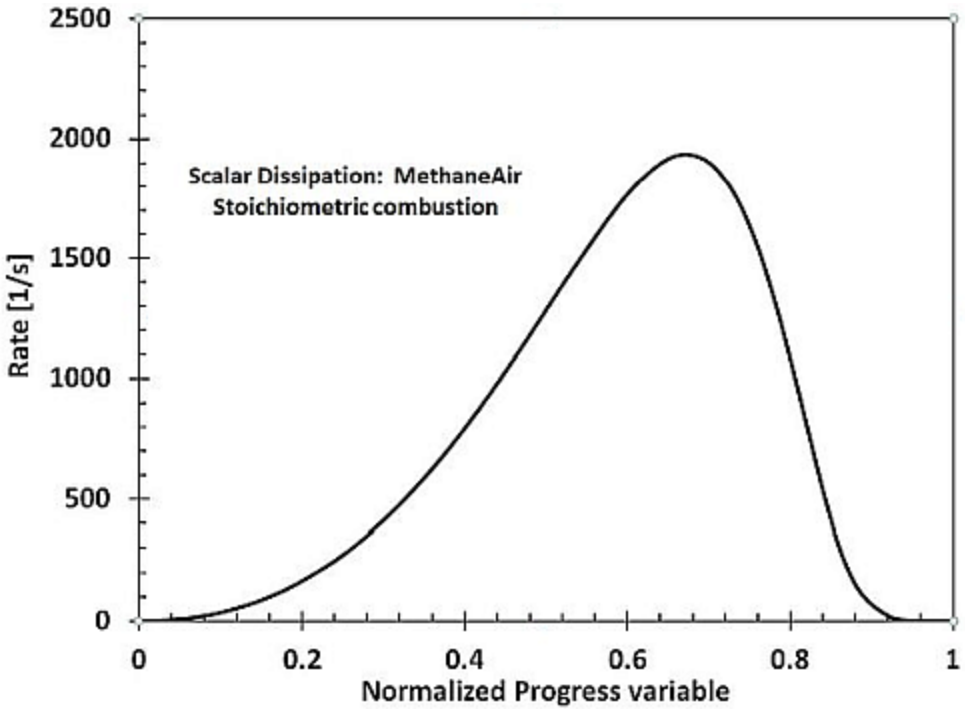
The maximum value of the scalar dissipation $\chi_{\max}^{sto}$ in Equation 8.112 (p. 347) is provided as a user input and is approximately calculated using the values of the flame thickness. The distribution of the scalar dissipation with respect to the reaction progress is modeled as a symmetric profile around the progress variable value c = 0.5 that decays exponentially towards the boundaries where c=0 or c =1.

Both approximations compromise the accuracy of the premixed FGM solution. If the flamelet equations are solved in the physical space, the scalar dissipation field is computed as part of the solution, and no assumption such as the shape of the scalar dissipation is made. Ansys Fluent allows you to generate the premixed FGM by solving flamelets in physical space using the Ansys Chemkin premixed flamelet generator. The governing equations and the method for solving laminar premixed flames are described in 1-D Premixed Laminar Flames in the *Chemkin Theory Manual*.

Figure 8.17: The Scalar Dissipation Rate Along The Normalized Reaction Progress Variable (p. 349) shows a distribution of the scalar dissipation for a stoichiometric methane air flamelet at 1 atm pressure calculated by the Ansys Chemkin premixed flamelet generator. The profile of the scalar dissipation exhibits an asymmetric peak distribution, which is consistent with the experimental results.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Figure 8.17: The Scalar Dissipation Rate Along The Normalized Reaction Progress Variable



8.3.2.2.3. Diffusion FGMs

For turbulent partially-premixed flames that are predominantly non-premixed, diffusion FGMs are a better representation of the thermochemistry than premixed FGMs. An example of this is modeling CO emissions from a gas-turbine combustor where the primary combustion zone is quenched by rapid mixing with dilution air. If the outlet equivalence ratio is less than the flammability limit of a corresponding premixed flamelet, the premixed FGM will predict sub-equilibrium CO, even if the combustor is quenched ($\overline{c}<1$). A diffusion FGM, however, will better predict super-equilibrium CO for $\overline{c}<1$.

Diffusion FGMs are calculated in Ansys Fluent using the diffusion laminar flamelet generator as detailed in Flamelet Generation (p. 323). Steady diffusion flamelets are generated over a range of scalar dissipation rates by starting from a very small strain (0.01/s by default) and increasing this in increments (5/s by default) until the flamelet extinguishes. The diffusion FGM is calculated from the steady diffusion laminar flamelets by converting the flamelet species fields to reaction progress, c (see Equation 8.106 (p. 344)). As the strain rate increases, the flamelet chemistry departs further from chemical equilibrium and c decreases from unity towards the extinction reaction, $c_{extinction}$.

Once the flame extincts, Ansys Fluent uses the time-history starting from the last successfully generated flamelet to obtain the unstable flamelets and, therefore, complete the entire FGM.

The Automated grid refinement (AGR) can be used when generating steady diffusion laminar flamelets. In the context of the diffusion FGM model, the gird in the mixture fraction space is refined. For information about the AGR, see Steady Diffusion Flamelet Automated Grid Refinement (p. 326).

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

8.3.2.2.4. Nonadiabatic Flamelet Generated Manifold (FGM)

The FGM manifold is generated by different types of flamelets (premixed or diffusion) as described in Flamelet Generated Manifold (FGM) Model (p. 345). The flamelets that constitute the manifold are generated with the adiabatic assumption. The generation process uses a single value of fuel and oxidizer inlet temperatures and, therefore, a single value of enthalpy at the fuel and oxidizer boundaries. The reference or representative temperature at the fuel and oxidizer inlets is used as fuel and oxidizer temperatures. The nonadiabatic PDF is generated from adiabatic flamelets assuming that the species composition does not depend on heat loss, and only physical properties (such as density and specific heat) are temperature-dependent. This assumption is reasonable for a large number of cases where the heat loss (and therefore the enthalpy variation) inside the combustor is a small fraction of reference enthalpy at which the adiabatic flamelets have been generated.

However, there are many scenarios where the enthalpy variation inside the combustor is significant. Examples of such cases involve:

- Combustor walls at much lower temperature than flame temperature, which leads to the high heat loss through the walls
- Multiple fuel and oxidizer inlets with different temperatures
- Post flame quenching
- Participating medium such as spray involving heat transfer with gas phase
- Participating medium (such as soot) that enhances heat transfer
- Temperature-sensitive species (such as NOx)

In such cases, the enthalpy changes within the system are not negligible, and therefore there is a considerable temperature variation inside the combustor. For such applications, the assumption of adiabatic flamelets can compromise the solution accuracy. To improve the fidelity of modeling combustion physics for these applications, the effect of heat loss or gain on the species composition should be considered. This can be achieved by generating flamelets with heat loss and gain.

Nonadiabatic Flamelet Generation

When generating flamelets, heat loss or gain can be modeled in different ways:

- including a heat sink term in the flamelet equations
- including the radiative heat loss in the flamelet equations
- generating flamelets with multiple reference conditions of temperature and enthalpies

The choice of the optimal method for flamelet generation is determined by the following factors:

- Computational efficiency of the manifold generation and optimal size of manifold
- Unique definition of each point in the manifold by reduced set of scalars (such as mixture fraction, progress variable and enthalpy) solved in the CFD solution.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

To model the effects of heat loss on the species composition, Ansys Fluent solves flamelets for different ranges of reference temperature and enthalpy conditions. The heat loss is modeled by generating successive flamelets with decreasing enthalpies of the fuel and oxidizer boundaries. For each enthalpy range, the flamelets can be efficiently generated using a freely propagating flame solved in physical space as described in Premixed FGMs in Physical Space (p. 348).

However, high heat loss (and therefore low enthalpy) values at fuel/oxidizer inlets can give unphysically low temperatures at the inlets, resulting in inaccurate flamelets. Therefore, the freely propagating flames can be used to model only low heat losses, and an alternative strategy is required to mimic high heat loss. Burner-stabilized flames described in Pre-mixed Burner-stabilized Stagnation Flame in the *Chemkin Theory Manual* is one of such approaches that allows modeling flamelets with high heat loss.

Ansys Fluent uses an optimized flamelet generation strategy, in which the nonadiabatic flamelets are generated in physical space by the Chemkin premix flamelet generator using a combination of the 1d premix freely propagating flame and the premixed burner stabilized flame. The Chemkin premix flamelet generator automatically chooses the solution method and the transition from one solution method to another based on the inlet species composition and enthalpy at the inlets.

For more information about these approaches, see the following sections in the Chemkin Theory Manual:

- 1-D Premixed Laminar Flames
- Pre-mixed Burner-stabilized Stagnation Flame in the *Chemkin Theory Manual*

8.3.2.3. FGM Turbulent Closure

The reaction progress variable is defined as a normalized fraction of product species (see Equation 8.107 (p. 346)), namely $c=\frac{Y_c}{Y_c^{eq}}$. Note that the denominator, Y_c^{eq}, is only a function of the local mixture fraction. When the **Flamelet Generated Manifold** model is enabled, Ansys Fluent solves a transport equation for the un-normalized progress variable, Y_c, and not the normalized progress variable, c. This has two advantages. Firstly, since there are usually no products in the oxidizer stream, Y_c^{eq} is zero and c is undefined here (in other words, burnt air is the same as unburnt air). This can lead to difficulties in specifying oxidizer boundary conditions where oxidizer is mixed into unburnt reactants before the flame, as well as into burnt products behind the flame. Solving for Y_c avoids these issues and the solution is independent of the specified boundary value of c for pure oxidizer inlets. The second advantage of solving for Y_c is that flame quenching can be modeled naturally. Consider a burnt stream in chemical equilibrium ($c=1$) that is rapidly quenched with an air jet. Since the c equation in Ansys Fluent does not have a source term dependent on changes in mixture fraction, c remains at unity and hence the diluted mixture remains at chemical equilibrium. Solving for Y_c can capture quenching since Y_c^{eq} changes with mixture fraction, and the normalized reaction progress c is correctly less than unity after mixing. To model these two effects with the normalized c equation, additional terms involving derivatives and cross-derivatives of mixture fraction would be required [76] (p. 1057). However, these terms do not appear in the transport equation for the density-weighted un-normalized progress variable $\widetilde{Y}_c$:

$$\frac{\partial\left(\rho\widetilde{Y}_c\right)}{\partial t}+\nabla\cdot\left(\rho\vec{v}\widetilde{Y}_c\right)=\nabla\cdot\left(\left(\frac{k}{C_p}+\frac{\mu_t}{Sc_t}\right)\nabla\widetilde{Y}_c\right)+\overline{S}_{Y_c} \tag{8.113}$$

where k is laminar thermal conductivity of the mixture, C_p is the mixture specific heat.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Typically, in the FGM model, the mean source term $\overline{S}_{Y_c}$ is modeled as:

$$\overline{S}_{Y_c}=\overline{\rho}\iint S_{FR,Y_c}(c,f)P(c,f)dcdf=\overline{S}_{FR} \tag{8.114}$$

where P is the joint PDF of reaction-progress (c) and mixture fraction (f), and S_{FR} is the Finite-Rate flamelet source term from the flamelet library. Similar to Equation 8.88 (p. 339), the mean source term S_{FR} is multiplied by a wall damping constant α_w at the wall boundaries. The default value of α_w is unity.

The source term $\overline{S}_{Y_c}$ determines the turbulent flame position. Errors in both the approximation of the Y_c variance, as well as the assumed shape Beta PDF, can cause inaccurate flame positions.

Ansys Fluent has another option for modeling turbulence-chemistry interaction for the source term, in which the closure is based on the turbulent flame speed:

$$\overline{S}_{Y_c}=\rho_u S_t\left|\nabla \overline{Y}_c\right|=\overline{S}_{TFS} \tag{8.115}$$

which is essentially the same source term when using the chemical equilibrium and diffusion flamelet partially-premixed models. An advantage of using a turbulent flame speed is that model constants can be calibrated to predict the correct flame position. In contrast, there are no direct parameters to control the FGM finite-rate source (Equation 8.114 (p. 352)), and hence the flame position.

The joint PDF, $P(c,f)$ in Equation 8.114 (p. 352), is specified as the product of two beta PDFs. The beta PDFs require second moments (that is, variances). The variance of the un-normalized reaction progress variable is modeled either with a transport equation

$$\frac{\partial\left(\rho\widetilde{Y_c''^2}\right)}{\partial t}+\nabla\cdot\left(\rho\vec{v}\widetilde{Y_c''^2}\right)=\nabla\cdot\left(\left(\frac{k}{C_p}+\frac{\mu_t}{Sc_t}\right)\nabla\widetilde{Y_c''^2}\right)+c_\varphi\frac{\mu_t}{Sc_t}\left|\nabla\tilde{Y}_c\right|^2-\frac{\rho c_\varphi}{\tau_{turb}}\widetilde{Y_c''^2} \tag{8.116}$$

where c_φ=2.0,

or with an algebraic expression

$$\widetilde{Y_c''^2}=c_{\text{var}}\frac{l_{turb}^2}{Sc_t}\left(\nabla\tilde{Y}_c\right)^2 \tag{8.117}$$

where l_{turb} is the turbulence length scale and c_{var} is a constant with a default value of 0.1.

For the SBES turbulence model, a blending formulation is used for the variance of the un-normalized reaction progress variable in the following way:

$$\widetilde{Y_c''^2}=\widetilde{Y_{c,RANS}''^2}f_{SDES}+\left(1-f_{SDES}\right)\widetilde{Y_{c,LES}''^2} \tag{8.118}$$

where subscripts $RANS$ and LES correspond to the RANS and LES modeling regions, respectively, and f_{SDES} is the shielding function defined in Equation 4.297 (p. 117).

8.3.2.3.1. Scalar Transport with FGM Closure

In the FGM model, the species concentration is obtained from reduced variables, such as the mixture fraction, progress variable, and their variances. Since the definition of the progress variable is predominantly based on the global species, the progress variable tracks the evolution of the major species and heat release reactions. In a reacting mixture, however, not all species evolve at the same rate due to the differences in their time scales. For example, the global progress variable may approach the unity, which marks the completion of a reaction. Meanwhile, a slow forming species, such as nitric oxide NO, may still be evolving. Therefore, mapping that is per-

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

formed solely on the basis of the global progress variable may lead to inaccurate results for the slow-forming species.

The progress variable is a reduced-order modeling approach that is based on a representative time scale of a reacting mixture. In reality, different species within a reacting mixture may have large disparities in their reaction time scales. Reduced-order models cannot capture these disparities. Solving for multiple separate transport equations for slow-forming species is one of the alternative approaches that will account for differences in the time scales of species evolution. Within the framework of the FGM model, Ansys Fluent provides an option to solve generic transport scalar equations, in addition to the global progress variable equation. These generic scalar transport equations can be solved for minor species that are low in concentration and, therefore, have no feedback on the main flow field, and that have time scales much slower than those of the major species in the mixture.

The transport equation of a generic scalar solved with the FGM model in Ansys Fluent has the following form:

$$\frac{\partial\left(\rho\widetilde{\varphi}\right)}{\partial t}+\nabla.\left(\rho\vec{v}\widetilde{\varphi}\right)=\nabla.\left(\left(\frac{k}{C_p}+\frac{\mu_t}{Sc_t}\right)\nabla\widetilde{\varphi}\right)+\overline{\rho}\overline{S}_{\varphi} \tag{8.119}$$

In this equation, the scalar $\widetilde{\varphi}$ that is to be transported can be a mass fraction of any species in the mixture. The last term is the reaction source term. It represents the net production rate of the scalar $\widetilde{\varphi}$ and is computed as the difference between its forward reaction rates and revere reaction rates.

The forward reaction rates of the scalar $\widetilde{\varphi}$ are not dependent on the value of $\widetilde{\varphi}$. Therefore, they can be pre-tabulated and stored in the PDF table. On the other hand, the reverse reaction rates of the scalar are dependent on the concentration of $\widetilde{\varphi}$. Therefore, the pre-tabulated values for the reverse reaction rates need to be corrected based on the solved transport scalar equation [267] (p. 1068).

In Ansys Fluent, the reverse reaction rates are first calculated using the prevailing values of the scalar $\widetilde{\varphi}$, and then they are stored in the PDF table. At the run time, the reverse reaction rates are scaled using the value of the solved transport scalar $\widetilde{\varphi}$ in the following way:

$$\overline{S}_{\varphi}=\overline{S}_{\varphi,fwd}-\overline{S}_{\varphi,rev}\frac{\widetilde{\varphi}}{\widetilde{\varphi}_{PDF}} \tag{8.120}$$

where $\overline{S}_{\varphi,fwd}$ and $\overline{S}_{\varphi,rev}$ are the forward and revere reaction rates, respectively, and $\widetilde{\varphi}_{PDF}$ is an averaged value of the transported scalar stored in the PDF table. This methodology allows the consumption of the scalar $\widetilde{\varphi}$ to respond to the slow evolution of the scalar solved.

The scalar transport model assumes that all the species, except for those solved as scalars, will occur over relatively short time scales. The scalar solved in Equation 8.119 (p. 353) is treated as a passive scalar, with no impact on the flow field, temperature, or mixture properties. Therefore, the scalar transport option should be used only for species that are low in concentration and that also have a slow formation rate. A typical example of such a species (for which a mass fraction can be solved as a scalar) is nitric oxide (NO). In the case of multiple scalar equations, the scalar

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

equation for each species is solved independently, and the concentration of the remaining species is taken from the PDF tables.

Note:

The scalar equations are treated as passive scalars that do not provide feedback to the flow equations or mixture properties. Therefore, for steady-state simulations, these equations can be solved in a post-processing mode, as for example, when modeling pollutant formation in Ansys Fluent.

8.3.2.4. Calculation of Mixture Properties

The flamelets generated using the premixed or diffusion FGM model store instantaneous species mass fractions and temperature as a function of the local mixture fraction f and the progress variable c. The mean thermochemical properties of the mixture are determined by averaging the instantaneous thermochemical property values using the modeled PDF $P(c,f)$ as:

$$\widetilde{\varphi}=\iint \varphi(c,f)P(c,f)dcdf \tag{8.121}$$

where $\widetilde{\varphi}$ denotes the species mass fraction or temperature from the flamelet files. In addition to temperature and species mass fraction, other mixture properties, such as specific heat, density, and molecular weight, can also be evaluated using Equation 8.121 (p. 354). The average mixture properties for an adiabatic system are represented as a function of mean mixture fraction $\widetilde{f}$, mean progress variable $\tilde{c}$, and their variances f'' and c'' and are stored inside a four-dimensional PDF table:

$$\widetilde{\varphi}=\widetilde{\varphi}(\widetilde{f},f'',\tilde{c},c'') \tag{8.122}$$

Non-adiabatic Extension for Average Mixture Properties Calculation

Similar to Non-Adiabatic Extensions of the Non-Premixed Model (p. 305), the fluctuations of enthalpy are ignored with the non-adiabatic extension to the PDF model. The average mixture properties for non-adiabatic partially premixed combustion using FGM can then be calculated as:

$$\widetilde{\varphi}=\iint \varphi(c,f,\widetilde{H})P(c,f)dcdf \tag{8.123}$$

where $\widetilde{H}$ is the mean enthalpy.

For non-adiabatic systems, each of the average mixture properties is a function of five independent variables:

$$\widetilde{\varphi}=\widetilde{\varphi}(\widetilde{f},f'',\tilde{c},c'',\widetilde{H}) \tag{8.124}$$

Calculation of average mixture properties using Equation 8.123 (p. 354) and Equation 8.124 (p. 354) requires a five-dimensional PDF table, which places an enormous demand on memory and could be computationally expensive. In order to optimize the run-time memory requirements and computational cost, the non-adiabatic PDF tables are generated with the following assumptions:

- Species mass fractions are not sensitive to the enthalpy change
- The mixture-averaged properties other than species mass fractions are computed using average progress variable

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

With these assumptions, the average species mass fraction and, therefore, molecular weight can be calculated using Equation 8.122 (p. 354), ignoring the enthalpy level. The other mixture properties, such as temperature, specific heat, and density, account for the enthalpy changes and are estimated as:

$$\overline{\varphi}=\iint \varphi\left(\tilde{c},f,\widetilde{H}\right)P\left(f\right)df \tag{8.125}$$

$$\overline{\varphi}=\overline{\varphi}\left(\widetilde{f},f'',\tilde{c},\widetilde{H}\right) \tag{8.126}$$

8.3.2.5. Calculation of Unburnt Properties

For FGM models, all properties, burnt or unburnt, are computed from the PDF tables as described in Equation 8.125 (p. 355) and Equation 8.126 (p. 355). However, for non-FGM models, unburnt properties are computed as described in this section.

Turbulent fluctuations are neglected for the unburnt mixture, so the mean unburnt scalars, $\overline{\phi}_u$, are functions of $\overline{f}$ only. The unburnt density, temperature, specific heat, and thermal diffusivity are fitted in Ansys Fluent to third-order polynomials of $\overline{f}$ using linear least squares:

$$\overline{\phi}_u=\sum_{n=0}^{3}c_n\overline{f}^n \tag{8.127}$$

Since the unburnt scalars are smooth and slowly-varying functions of $\overline{f}$, these polynomial fits are generally accurate. Access to polynomials is provided in case you want to modify them.

When the secondary mixture fraction model is enabled, the unburnt density, temperature, specific heat, thermal diffusivity, and laminar flame speed are calculated as follows: polynomial functions are calculated for a mixture of pure primary fuel and oxidizer, as described above, and are a function of the mean primary mixture fraction, $\overline{f}$. Similar polynomial functions are calculated for a mixture of pure secondary fuel and oxidizer, and are a function of the normalized secondary mixture fraction, $\overline{p}$. The unburnt properties in a cell are then calculated as a weighted function of the mean primary mixture fraction and mean secondary normalized mixture fraction as,

$$\phi_u=\frac{\overline{f}}{\overline{f}+\overline{p}}\phi_u^{primary}+\frac{\overline{p}}{\overline{f}+\overline{p}}\phi_u^{secondary} \tag{8.128}$$

8.3.2.6. Laminar Flame Speed

The premixed models require the laminar flame speed (see Equation 8.77 (p. 337)), which depends strongly on the composition, temperature, and pressure of the unburnt mixture. For adiabatic perfectly premixed systems as in Premixed Combustion (p. 332), the reactant stream has one composition, and the laminar flame speed is constant throughout the domain. However, in partially premixed systems, the laminar flame speed will change as the reactant composition (equivalence ratio) changes, and this must be taken into account.

Accurate laminar flame speeds are difficult to determine analytically, and are usually measured from experiments or computed from 1D simulations. For the partially-premixed model, in addition to the laminar flame speed model options described in Laminar Flame Speed (p. 342), namely constant, user-defined function, and Metghalchi-Keck, Ansys Fluent offers the following methods:

- prepdf-polynomial

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

The prepdf-polynomial method is based on fitted curves obtained from numerical simulations of the laminar flame speed [216] (p. 1065). These curves were determined for hydrogen (H_2), methane (CH_4), acetylene, (C_2H_2), ethylene, (C_2H_4), ethane, (C_2H_6), and propane (C_3H_8) fuels. They are valid for inlet compositions ranging from the lean limit through unity equivalence ratio (stoichiometric), for unburnt temperatures from 298 K to 800 K, and for pressures from 1 bar to 40 bars.

- laminar-flame-speed-computed

  For the laminar-flame-speed-computed method, the laminar flame speed is computed by the Ansys Chemkin premix flame reactor model using 1-D premixed laminar flames in physical space. The calculations are based on the boundary conditions provided for the fuel and oxidizer. The governing equations and the method for solving 1-D premixed laminar flames are described in 1-D Premixed Laminar Flames in the *Chemkin Theory Manual*.

  Although the laminar-flame-speed-computed method is a generalized method that can be used for any fuel and operating conditions, it is particularly useful when a blend of fuels with significantly different flame characteristics (for example, a fuel blend of hydrogen and natural gas) is simulated.

- laminar-flame-speed-library

  In the laminar-flame-speed-library method, the laminar flame speed is calculated using the table lookup. Ansys Fluent comes installed with pre-built laminar flame-speed tables for the following most common hydrocarbon fuels:

a2ch3	c2h5oh	c2h6	c3h8	c4h8o1-4	c4h10	c5h10-1	c5h10-2
c6h5c2h5	c6h5c3h7	c6h5ch3	c6h12-1	c6h12-2	c6h12-3	ch3och3	ch3oh
ch4	decalin	etfe	h2	hmn	ic4h10	ic5h12	ic8h18
mch	mtbe	m-xylene	nc4h9oh	nc5h12	nc6h14	nc7h16	nc12h26
nc14h30	nc16h34	p-xylene	tmb124	bc5h10	c2h2	c2h4	c3h4-a
c3h4-p	c4h6	co	cy13pd	ic6h14	mb	mb2d	nc10h22
o-xylene	c3h6	ic12h26	nc9h20	chx			

  For additional information about pre-built laminar flame-speed tables, see Flame-Speed Tables Installed with Ansys Forte in the *Ansys Forte User's Guide*.

  The laminar flame-speed tables were generated using the Ansys Chemkin [102] (p. 1058) Flame-speed Table Generator. In the generator, the laminar flame speed was computed by considering a freely propagating 1D adiabatic premixed flame. For each fuel, a detailed chemical reaction mechanism was abstracted from a master mechanism with over 3000 species. The generator conducted numerous numerical simulations in order to cover a wide range of pressure, temperature, equivalent ratio, and EGR conditions. The calculated flame speed values were then tabulated. The conditions covered for the built-in tables are summarized in the following table:

Parameter varied	Range of values
Equivalence ratio	0.4-2.0
Pressure (bar)	1-150
Unburnt temperature (K)	300-1200
Dilution(EGR) rate by mass	0-40%

  For Hydrogen and CO, the equivalence ratio tabulation spans 0.3 to 6.0. For all other fuels, the maximum value is 2.0 as denoted in the above table.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

If an unburnt mixture contains multiple fuels, a mixing law is used to compute the mixture laminar flame speed.

For all these methods, Ansys Fluent fits the curves to a piecewise-linear polynomial. Mixtures leaner than the lean limit or richer than the rich limit will not burn and have zero flame speed. The required inputs are values for the laminar flame speed at 20 mixture fraction ($\overline{f}$) points.

For non-adiabatic simulations, such as heat transfer at walls or compressive heating, the unburnt mixture temperature may deviate from its adiabatic value. The piecewise-linear function of mixture fraction is unable to account for this effect on the laminar flame speed. You can include non-adiabatic effects on the laminar flame speed by enabling **Non-Adiabatic Laminar Flame Speed**, which tabulates the laminar speeds in the PDF table by evaluating the curve fits from [216] (p. 1065) at the enthalpy levels in the PDF table. Note that the tabulated mean laminar flame speed accounts for fluctuations in the mixture fraction.

Important:

In the prepdf-polynomial method, the flame speed fits are accurate for air mixtures with pure fuels of H_2, CH_4, C_2H_2, C_2H_4, C_2H_6, and C_3H_8. If an oxidizer other than air or a different fuel is used, or if the unburnt temperature or pressure is outside the range of validity, then the curve fits will be incorrect. Although Ansys Fluent defaults to a methane-air mixture, the laminar flame speed polynomial and the rich and lean limits are most likely incorrect for your specified fuel/oxidizer and unburnt temperature/pressure conditions.

8.3.2.7. Strained Laminar Flame Speed

In the Flamelet Generated Manifold (FGM) model, the reaction source term for the progress variable equation can be modeled using either the finite rate source as in Equation 8.114 (p. 352) or the turbulent flame speed closure as in Equation 8.115 (p. 352). When the turbulent flame speed closure is employed, Ansys Fluent tabulates the laminar flame speed using either prepdf polynomials or the flame speed library as described in Laminar Flame Speed (p. 355). The flame speed computed from different methods described in Laminar Flame Speed (p. 355) is obtained by solving one-dimensional premixed freely propagating flame configurations. The flame speed is then expressed as a polynomial function of the mixture fraction.

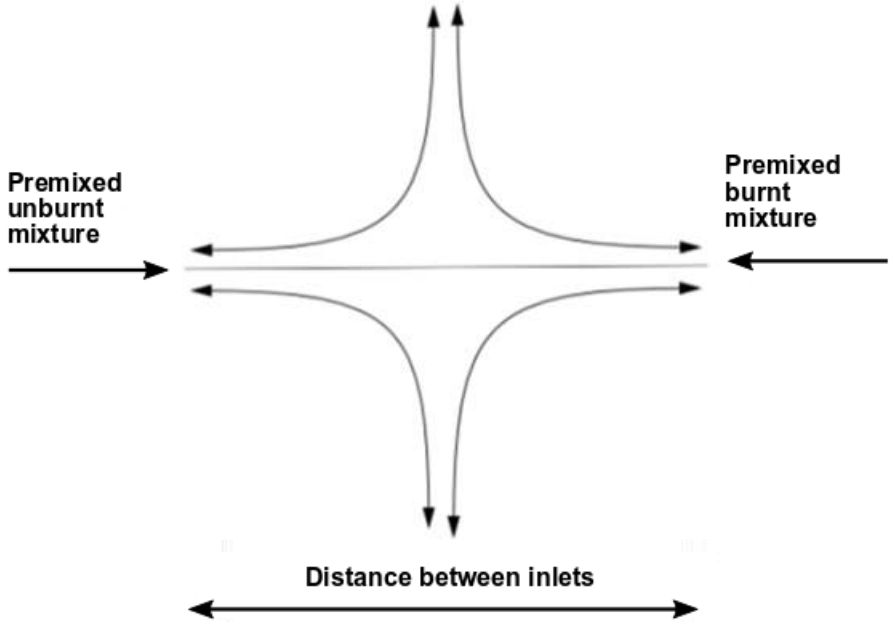
In most practical combustion systems, the flames experience strain due to various factors such as turbulence, swirl, geometry, and so on. The impact of the strain rate on the flame can be neglected for a wide range of applications, and, therefore, the flame speed can be computed using the freely propagating flames with reasonable accuracy. However, the impact of the strain on the flame is magnified in certain scenarios such as the lean blow-off limit, highly swirling flames with lean premixed conditions, and so on. For these applications, using the unstrained flame speed can often over-predict the flame speed and may lead to inaccurate prediction of critical combustion characteristics. In such cases, the accuracy of the combustion modeling can be improved if the impact of the strain rate on the flame speed is taken into account.

Such effects are considered in the strained flame speed model in Ansys Fluent. In this model, the strained flame speed is tabulated as a function of the strain rate and mixture fraction. For each strain rate and mixture fraction, the strained flame speed is calculated using one-dimensional premixed opposed-flow strained flamelets. The computational domain of this configuration contains two opposed-flow inlets and one-dimensional solution domain in physical space. The configuration

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

schematics for flamelet generation is shown in Figure 8.18: Premix Opposed Flow Configuration for the Strained Flame Speed (p. 358).

Figure 8.18: Premix Opposed Flow Configuration for the Strained Flame Speed



The premixed opposed-flow strained flamelets are solved in physical space using the Ansys Chemkin Oppdif solver. The details of the Oppdif solver are provided in Opposed-flow and Stagnation Flames in the *Chemkin Theory Manual*.

The strained flamelets use the following boundary conditions for the two inlets:

- Premixed unburnt mixture inlet: A premixed unburnt mixture with a specified mixture fraction at one of the inlets
- Premixed burnt mixture inlet: A composition of a fully burnt mixture with the same mixture fraction as for the premixed unburnt mixture at the second inlet

Flamelet Generation and Flame Speed Calculations

The strain rate a (1/s) is computed as

$$a=\frac{\left(v_b+v_U\right)}{d} \tag{8.129}$$

where

v_b = burning rate of fuel

v_U = velocity at the unburnt mixture inlet

d = distance between the two inlets

If both inlet jets have equal momentum, then

$$\rho_u v_u=\rho_b v_b \tag{8.130}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

where ρ_u and ρ_b are the densities of the unburnt and burnt mixtures, respectively.

From Equation 8.129 (p. 358) and Equation 8.130 (p. 358), the strain rate can be obtained as:

$$a = \frac{v_u\left(\frac{r_n}{r_b}+1\right)}{d} \tag{8.131}$$

The strained flamelets are generated to increase the strain rate a to its maximum value. For a given strain rate, both inlet velocities are computed using Equation 8.130 (p. 358) and Equation 8.131 (p. 359). The strained flame speed S_c for a given mixture fraction and strain rate is computed as:

$$S_c = \frac{1}{\rho_u y_{fuel}} \int_{-\infty}^{\infty} \dot{\omega} dx \tag{8.132}$$

where $\dot{\omega}$ is the burning rate of fuel, and y_{fuel} is the mass fraction of the fuel. Prior to the CFD solution, the strained flame speed S_c is computed as a function of the strain rate a and mixture fraction f and stored as a table.

During the CFD solution, the strained flame speed is interpolated as a function of the mixture fraction and computed strain rate using the stored data. It is then used in the turbulent flame speed closure in Equation 8.115 (p. 352):

Strain Rate Calculation in the CFD Solution

The solution of the strained flamelets is used to compute the strained flame speed, which is tabulated as a function of the strain rate and mixture fraction. The mixture fraction is obtained from the solution of the transport equation as described in Premixed FGMs in Physical Space (p. 348). The total strain rate in the CFD solution is computed using the local flow conditions as a sum of the strain rate due to mean flow and due to turbulence [643] (p. 1090):

$$a_{total} = a_{mean} + a_{turb} \tag{8.133}$$

where the strain rate due to mean flow a_{mean} is computed as:

$$a_{mean} = \left(\nabla \cdot \vec{v}\right) - \vec{n} \cdot \left(\vec{n} \cdot \nabla \vec{v}\right) \tag{8.134}$$

and the strain rate due to the turbulent flow is modeled as [118] (p. 1059):

$$a_{turb} = \frac{\epsilon}{k}\left[\left(1-\alpha_0\right) + \alpha_0 \Gamma_k\right] \tag{8.135}$$

In the above equations,

$\vec{v}$ = flow velocity in the control volume

$\vec{n}$ = unit normal to the flame surface; $\vec{n} = \frac{\nabla C}{|\nabla C|}$, where C is the progress variable

ϵ = turbulent dissipation rate

k = turbulent kinetic energy

Γ_k = intermediate turbulent net flame stretch (ITNFS) term

α_0 = user-speecified coefficient that determines the weight of the term Γ_k relative to a straightforward turbulent time scale (default = 1)

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

For RANS-based turbulence models, Γ_k=1.

For LES models, Γ_k is computed as:

$$\log_{10}(\Gamma_k)=-\frac{1}{s+0.4}\exp(-(s+0.4))+(1-\exp(-(s+0.4)))\left(\sigma_1\left(\frac{u'}{U_l}\right)s-0.11\right) \tag{8.136}$$

with

$$s=\log_{10}\left(\frac{l_t}{\delta_l^0}\right) \tag{8.137}$$

and

$$\sigma_1\left(\frac{u'}{U_l}\right)=\frac{2}{3}\left(1-\frac{1}{2}\exp\left(-\left(\frac{u'}{U_l}\right)^{1/3}\right)\right) \tag{8.138}$$

In Equation 8.136 (p. 360) thru Equation 8.138 (p. 360), the following notations are used:

u' = turbulent velocity fluctuation

U_l = laminar flame speed

l_t = integral turbulent length scale

δ_l^0 = laminar flame thickness

8.3.2.7.1. Strained Non-Adiabatic Flame Speed

Many combustion devices involve significant heat loss through the walls. In such cases, the flame speed is not only dependent on the strain rate, but also sensitive to the heat loss. To account for the impact of heat loss on strained flamelets, Ansys Fluent provides a non-adiabatic extension of the strained flame speed. In this approach, the flame speed is computed using the premixed opposed flow configuration (see Figure 8.18: Premix Opposed Flow Configuration for the Strained Flame Speed (p. 358)) with a different heat loss parameter applied on the premixed burnt mixture side inlet.

The heat loss is caused by cooling the burned products, and the heat loss parameter is defined as:

$$\psi=1-\frac{T_{burned}}{T_{adia,\,burned}} \tag{8.139}$$

where T_{burned} is the temperature achieved by cooling the burned products, and $T_{adia,burned}$ is the temperature of the premixed burnt mixture without any heat loss.

For the non-adiabatic strained flame speed calculation, the strained flamelets are generated for different levels of the heat loss parameter. ψ =0 corresponds to no heat loss, and therefore gives the strained adiabatic flame speed. The non-adiabatic strained flame speed is stored in a three-dimensional table as a function of the mixture fraction (f), strain rate (χ), and heat loss parameter (ψ):

$$S_c=S_c(f,\chi,\psi) \tag{8.140}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

During the CFD solution, the heat loss parameter (ψ_{CFD}) is calculated for each cell volume as a ratio of temperatures for cell enthalpy (H) and adiabatic enthalpy (H_{adia}) obtained for the same mixture fraction (f) and progress variable (C):

$$\psi_{CFD}=\frac{T\left(f,C,H\right)}{T\left(f,C,H_{adia}\right)} \tag{8.141}$$

The strained flame speed from the PDF table is then interpolated using the mixture fraction (f), strain rate (χ) and cell heat loss parameter (ψ_{CFD}).

8.3.2.8. Generating PDF Lookup Tables Through Automated Grid Refinement

In both the diffusion and premixed FGM models, the automated grid refinement (AGR) can be used for generating the PDF lookup table. For further details about AGR, see Generating Lookup Tables Through Automated Grid Refinement (p. 312). In the current implementation, the grid points used in the mixture fraction and reaction-progress space are fixed, and they are taken from the flamelet calculation. The grid refinement procedure is conducted in the mixture fraction variance, reaction progress variance, and mean enthalpy space.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information
of ANSYS, Inc. and its subsidiaries and affiliates.

Chapter 9: Pollutant Formation

This chapter discusses the theory behind the models available in Ansys Fluent for modeling pollutant formation.

Information is presented in the following sections:

9.1. NOx Formation

9.2. Soot Formation

9.3. Decoupled Detailed Chemistry Model

For information about using the models in Ansys Fluent, see Modeling Pollutant Formation in the User's Guide.

9.1. NOx Formation

The following sections present the theoretical background of NO_x prediction. For information about using the NO_x models in Ansys Fluent, see Using the NOx Model in the User's Guide.

9.1.1. Overview

9.1.2. Governing Equations for NOx Transport

9.1.3. Thermal NOx Formation

9.1.4. Prompt NOx Formation

9.1.5. Fuel NOx Formation

9.1.6. NOx Formation from Intermediate N2O

9.1.7. NOx Reduction by Reburning

9.1.8. NOx Reduction by SNCR

9.1.9. NOx Formation in Turbulent Flows

9.1.1. Overview

NO_x emission consists of mostly nitric oxide (NO), and to a lesser degree nitrogen dioxide (NO_2) and nitrous oxide (N_2O). NO_x is a precursor for photochemical smog, contributes to acid rain, and causes ozone depletion. Thus, NO_x is a pollutant. The Ansys Fluent NO_x model provides a tool to understand the sources of NO_x production and to aid in the design of NO_x control measures.

9.1.1.1. NOx Modeling in Ansys Fluent

The Ansys Fluent NO_x model provides the capability to model thermal, prompt, and fuel NO_x formation, as well as NO_x consumption due to reburning in combustion systems. It uses rate models developed at the Department of Fuel and Energy at The University of Leeds in England, as well as

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

from the open literature. NO_x reduction using reagent injection, such as selective non-catalytic reduction (SNCR), can be modeled in Ansys Fluent, along with an N_2O intermediate model that has also been incorporated.

To predict NO_x emissions, Ansys Fluent solves a transport equation for nitric oxide (NO) concentration. When fuel NO_x sources are present, Ansys Fluent solves additional transport equations for intermediate species (HCN and/or NH_3). When the N_2O intermediate model is activated, an additional transport equation for N_2O will be solved. The NO_x transport equations are solved based on a given flow field and combustion solution. In other words, NO_x is postprocessed from a combustion simulation. It is therefore evident that an accurate combustion solution becomes a prerequisite of NO_x prediction. For example, thermal NO_x production doubles for every 90 K temperature increase when the flame temperature is about 2200 K. Great care must be exercised to provide accurate thermophysical data and boundary condition inputs for the combustion model. Appropriate turbulence, chemistry, radiation, and other submodels must be employed.

To be realistic, you can only expect results to be as accurate as the input data and the selected physical models. Under most circumstances, NO_x variation trends can be accurately predicted, but the NO_x quantity itself cannot be pinpointed. Accurate prediction of NO_x parametric trends can cut down on the number of laboratory tests, allow more design variations to be studied, shorten the design cycle, and reduce product development cost. That is truly the power of the Ansys Fluent NO_x model and, in fact, the power of CFD in general.

9.1.1.2. NOx Formation and Reduction in Flames

In laminar flames and at the molecular level within turbulent flames, the formation of NO_x can be attributed to four distinct chemical kinetic processes: thermal NO_x formation, prompt NO_x formation, fuel NO_x formation, and intermediate N_2O. Thermal NO_x is formed by the oxidation of atmospheric nitrogen present in the combustion air. Prompt NO_x is produced by high-speed reactions at the flame front. Fuel NO_x is produced by oxidation of nitrogen contained in the fuel. At elevated pressures and oxygen-rich conditions, NO_x may also be formed from molecular nitrogen (N_2) via N_2O. The reburning and SNCR mechanisms reduce the total NO_x formation by accounting for the reaction of NO with hydrocarbons and ammonia, respectively.

Important:

The NO_x models cannot be used in conjunction with the premixed combustion model.

9.1.2. Governing Equations for NOx Transport

Ansys Fluent solves the mass transport equation for the NO species, taking into account convection, diffusion, production, and consumption of NO and related species. This approach is completely general, being derived from the fundamental principle of mass conservation. The effect of residence time in NO_x mechanisms (a Lagrangian reference frame concept) is included through the convection terms in the governing equations written in the Eulerian reference frame. For thermal and prompt NO_x mechanisms, only the NO species transport equation is needed:

$$\frac{\partial}{\partial t}\left(\rho Y_{NO}\right)+\nabla\cdot\left(\rho\vec{v}Y_{NO}\right)=\nabla\cdot\left(\rho\mathcal{D}\nabla Y_{NO}\right)+S_{NO} \tag{9.1}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

As discussed in Fuel NOx Formation (p. 371), the fuel NO_x mechanisms are more involved. The tracking of nitrogen-containing intermediate species is important. Ansys Fluent solves a transport equation for the HCN, NH_3, or N_2O species, in addition to the NO species:

$$\frac{\partial}{\partial t}\left(\rho Y_{HCN}\right)+\nabla\cdot\left(\rho\vec{v}Y_{HCN}\right)=\nabla\cdot\left(\rho\mathcal{D}Y_{HCN}\right)+S_{HCN} \tag{9.2}$$

$$\frac{\partial}{\partial t}\left(\rho Y_{NH_3}\right)+\nabla\cdot\left(\rho\vec{v}Y_{NH_3}\right)=\nabla\cdot\left(\rho\mathcal{D}Y_{NH_3}\right)+S_{NH_3} \tag{9.3}$$

$$\frac{\partial}{\partial t}\left(\rho Y_{N_2O}\right)+\nabla\cdot\left(\rho\vec{v}Y_{N_2O}\right)=\nabla\cdot\left(\rho\mathcal{D}Y_{N_2O}\right)+S_{N_2O} \tag{9.4}$$

where Y_{HCN}, Y_{NH_3}, Y_{N_2O}, and Y_{NO} are mass fractions of HCN, NH_3, N_2O, and NO in the gas phase, and $\mathcal{D}$ is the effective diffusion coefficient. The source terms S_{HCN}, S_{NH_3}, S_{N_2O}, and S_{NO} are to be determined next for different NO_x mechanisms.

9.1.3. Thermal NOx Formation

The formation of thermal NO_x is determined by a set of highly temperature-dependent chemical reactions known as the extended Zeldovich mechanism. The principal reactions governing the formation of thermal NO_x from molecular nitrogen are as follows:

$$\mathrm{O+N_2\rightleftharpoons N+NO} \tag{9.5}$$

$$\mathrm{N+O_2\rightleftharpoons O+NO} \tag{9.6}$$

A third reaction has been shown to contribute to the formation of thermal NO_x, particularly at near-stoichiometric conditions and in fuel-rich mixtures:

$$\mathrm{N+OH\rightleftharpoons H+NO} \tag{9.7}$$

9.1.3.1. Thermal NOx Reaction Rates

The rate constants for these reactions have been measured in numerous experimental studies [68] (p. 1056), [186] (p. 1063), [447] (p. 1079), and the data obtained from these studies have been critically evaluated by Baulch et al. [48] (p. 1055) and Hanson and Salimian [233] (p. 1066). The expressions for the rate coefficients for Equation 9.5 (p. 365) – Equation 9.7 (p. 365) used in the NO_x model are given below. These were selected based on the evaluation of Hanson and Salimian [233] (p. 1066).

$$k_{f,1} = 1.8\times10^{8}e^{-38370/T} \qquad k_{r,1} = 3.8\times10^{7}e^{-425/T}$$

$$k_{f,2} = 1.8\times10^{4}Te^{-4680/T} \qquad k_{r,2} = 3.81\times10^{3}Te^{-20820/T}$$

$$k_{f,3} = 7.1\times10^{7}e^{-450/T} \qquad k_{r,3} = 1.7\times10^{8}e^{-24560/T}$$

In the above expressions, $k_{f,1}$, $k_{f,2}$, and $k_{f,3}$ are the rate constants for the forward reactions Equation 9.5 (p. 365) – Equation 9.7 (p. 365), respectively, and $k_{r,1}$, $k_{r,2}$, and $k_{r,3}$ are the corresponding reverse rate constants. All of these rate constants have units of m^3/mol-s.

The net rate of formation of NO via the reactions in Equation 9.5 (p. 365) – Equation 9.7 (p. 365) is given by

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$\frac{d[\mathrm{NO}]}{dt} = k_{f,1}[\mathrm{O}][\mathrm{N_2}] + k_{f,2}[\mathrm{N}][\mathrm{O_2}] + k_{f,3}[\mathrm{N}][\mathrm{OH}] - k_{r,1}[\mathrm{NO}][\mathrm{N}] - k_{r,2}[\mathrm{NO}][\mathrm{O}] - k_{r,3}[\mathrm{NO}][\mathrm{H}] \tag{9.8}$$

where all concentrations have units of mol/m^3.

To calculate the formation rates of NO and N, the concentrations of O, H, and OH are required.

9.1.3.2. The Quasi-Steady Assumption for [N]

The rate of formation of NO_x is significant only at high temperatures (greater than 1800 K) because fixation of nitrogen requires the breaking of the strong N_2 triple bond (providing dissociation energy of 941 kJ/mol). This effect is represented by the high activation energy of reaction Equation 9.5 (p. 365), which makes it the rate-limiting step of the extended Zeldovich mechanism. However, the activation energy for oxidation of N atoms is small. When there is sufficient oxygen, as in a fuel-lean flame, the rate of consumption of free nitrogen atoms becomes equal to the rate of its formation, and therefore a quasi-steady state can be established. This assumption is valid for most combustion cases, except in extremely fuel-rich combustion conditions. Hence the NO formation rate becomes

$$\frac{d[\mathrm{NO}]}{dt} = 2k_{f,1}[\mathrm{O}][\mathrm{N_2}] \frac{\left(1 - \frac{k_{r,1}k_{r,2}[NO]^2}{k_{f,1}[N_2]k_{f,2}[O_2]}\right)}{\left(1 + \frac{k_{r,1}[NO]}{k_{f,2}[O_2] + k_{f,3}[OH]}\right)} \quad (\mathrm{mol/m^3\text{-}s}) \tag{9.9}$$

9.1.3.3. Thermal NOx Temperature Sensitivity

From Equation 9.9 (p. 366) it is clear that the rate of formation of NO will increase with increasing oxygen concentration. It also appears that thermal NO formation should be highly dependent on temperature but independent of fuel type. In fact, based on the limiting rate described by $k_{f,1}$, the thermal NO_x production rate doubles for every 90 K temperature increase beyond 2200 K.

9.1.3.4. Decoupled Thermal NOx Calculations

To solve Equation 9.9 (p. 366), the concentration of O atoms and the free radical OH will be required, in addition to the concentration of stable species (that is, O_2, N_2). Following the suggestion by Zeldovich, the thermal NO_x formation mechanism can be decoupled from the main combustion process by assuming equilibrium values of temperature, stable species, O atoms, and OH radicals. However, radical concentrations (O atoms in particular) are observed to be more abundant than their equilibrium levels. The effect of partial equilibrium O atoms on NO_x formation rate has been investigated [442] (p. 1078) during laminar methane-air combustion. The results of these investigations indicate that the level of NO_x emission can be under-predicted by as much as 28% in the flame zone, when assuming equilibrium O-atom concentrations.

9.1.3.5. Approaches for Determining O Radical Concentration

There has been little detailed study of radicals concentration in industrial turbulent flames, but work [151] (p. 1061) has demonstrated the existence of this phenomenon in turbulent diffusion flames. Presently, there is no definitive conclusion as to the effect of partial equilibrium on NO_x formation

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

rates in turbulent flames. Peters and Donnerhack [512] (p. 1083) suggest that partial equilibrium radicals can account for no more than a 25% increase in thermal NO_x and that fluid dynamics has the dominant effect on the NO_x formation rate. Bilger and Beck [60] (p. 1056) suggest that in turbulent diffusion flames, the effect of O atom overshoot on the NO_x formation rate is very important.

To overcome this possible inaccuracy, one approach would be to couple the extended Zeldovich mechanism with a detailed hydrocarbon combustion mechanism involving many reactions, species, and steps. This approach has been used previously for research purposes [437] (p. 1078). However, high computational cost of such approaches limits their use in turbulent reacting flows.

To determine the O radical concentration, Ansys Fluent provides the equilibrium, partial equilibrium, and the predicted concentration approaches described below.

9.1.3.5.1. Method 1: Equilibrium Approach

The kinetics of the thermal NO_x formation rate is much slower than the main hydrocarbon oxidation rate, and so most of the thermal NO_x is formed after completion of combustion. Therefore, the thermal NO_x formation process can often be decoupled from the main combustion reaction mechanism and the NO_x formation rate can be calculated by assuming equilibration of the combustion reactions. Using this approach, the calculation of the thermal NO_x formation rate is considerably simplified. The assumption of equilibrium can be justified by a reduction in the importance of radical overshoots at higher flame temperature [150] (p. 1061). According to Westenberg [701] (p. 1093), the equilibrium O-atom concentration can be obtained from the expression

$$[\mathrm{O}] = k_p [\mathrm{O}_2]^{1/2} \tag{9.10}$$

With k_p included, this expression becomes

$$[\mathrm{O}] = 3.97 \times 10^5 T^{-1/2} [\mathrm{O}_2]^{1/2} e^{-31090/T} \quad \mathrm{mol/m^3} \tag{9.11}$$

where T is in Kelvin.

9.1.3.5.2. Method 2: Partial Equilibrium Approach

An improvement to method 1 can be made by accounting for third-body reactions in the O_2 dissociation-recombination process:

$$\mathrm{O_2 + M \rightleftharpoons O + O + M} \tag{9.12}$$

Equation 9.11 (p. 367) is then replaced by the following expression [690] (p. 1093):

$$[\mathrm{O}] = 36.64 T^{1/2} [\mathrm{O}_2]^{1/2} e^{-27123/T} \quad \mathrm{mol/m^3} \tag{9.13}$$

which generally leads to a higher partial O-atom concentration.

9.1.3.5.3. Method 3: Predicted O Approach

When the O-atom concentration is well predicted using an advanced chemistry model (such as the flamelet submodel of the non-premixed model), [O] can be taken simply from the local O-species mass fraction.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

9.1.3.6. Approaches for Determining OH Radical Concentration

Ansys Fluent uses one of three approaches to determine the OH radical concentration: the exclusion of OH from the thermal NO_x calculation, the partial equilibrium, and the use of the predicted OH concentration.

9.1.3.6.1. Method 1: Exclusion of OH Approach

In this approach, the third reaction in the extended Zeldovich mechanism (Equation 9.7 (p. 365)) is assumed to be negligible through the following observation:

$$k_2[O_2]_{eq} \gg k_3[OH]_{eq}$$

This assumption is justified for lean fuel conditions and is a reasonable assumption for most cases.

9.1.3.6.2. Method 2: Partial Equilibrium Approach

In this approach, the concentration of OH in the third reaction in the extended Zeldovich mechanism (Equation 9.7 (p. 365)) is given by [49] (p. 1055), [700] (p. 1093):

$$[\mathrm{OH}] = 2.129 \times 10^2 T^{-0.57} e^{-4595/T} [\mathrm{O}]^{1/2} [\mathrm{H_2O}]^{1/2} \quad \mathrm{mol/m^3} \tag{9.14}$$

9.1.3.6.3. Method 3: Predicted OH Approach

As in the predicted O approach, when the OH radical concentration is well predicted using an advanced chemistry model such as the flamelet model, [OH] can be taken directly from the local OH species mass fraction.

9.1.3.7. Summary

To summarize, the thermal NO_x formation rate is predicted by Equation 9.9 (p. 366). The O-atom concentration needed in Equation 9.9 (p. 366) is computed using Equation 9.11 (p. 367) for the equilibrium assumption, using Equation 9.13 (p. 367) for a partial equilibrium assumption, or using the local O-species mass fraction. You will make the choice during problem setup. In terms of the transport equation for NO (Equation 9.1 (p. 364)), the NO source term due to thermal NO_x mechanisms is

$$S_{thermal,NO} = M_{w,NO} \frac{d[NO]}{dt} \tag{9.15}$$

where $M_{w,NO}$ is the molecular weight of NO (kg/mol), and $d[NO]/dt$ is computed from Equation 9.9 (p. 366).

9.1.4. Prompt NOx Formation

It is known that during combustion of hydrocarbon fuels, the NO_x formation rate can exceed that produced from direct oxidation of nitrogen molecules (that is, thermal NO_x).

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

9.1.4.1. Prompt NOx Combustion Environments

The presence of a second mechanism leading to NO_x formation was first identified by Fenimore [171] (p. 1062) and was termed "prompt NO_x". There is good evidence that prompt NO_x can be formed in a significant quantity in some combustion environments, such as in low-temperature, fuel-rich conditions and where residence times are short. Surface burners, staged combustion systems, and gas turbines can create such conditions [41] (p. 1055).

At present, the prompt NO_x contribution to total NO_x from stationary combustors is small. However, as NO_x emissions are reduced to very low levels by employing new strategies (burner design or furnace geometry modification), the relative importance of the prompt NO_x can be expected to increase.

9.1.4.2. Prompt NOx Mechanism

Prompt NO_x is most prevalent in rich flames. The actual formation involves a complex series of reactions and many possible intermediate species. The route now accepted is as follows:

$$CH + N_2 \rightleftharpoons HCN + N \tag{9.16}$$

$$N + O_2 \rightleftharpoons NO + O \tag{9.17}$$

$$HCN + OH \rightleftharpoons CN + H_2O \tag{9.18}$$

$$CN + O_2 \rightleftharpoons NO + CO \tag{9.19}$$

A number of species resulting from fuel fragmentation have been suggested as the source of prompt NO_x in hydrocarbon flames (for example, CH, CH_2, C, C_2H), but the major contribution is from CH (Equation 9.16 (p. 369)) and CH_2, via

$$CH_2 + N_2 \rightleftharpoons HCN + NH \tag{9.20}$$

The products of these reactions could lead to formation of amines and cyano compounds that subsequently react to form NO by reactions similar to those occurring in oxidation of fuel nitrogen, for example:

$$HCN + N \rightleftharpoons N_2 + \ldots \tag{9.21}$$

9.1.4.3. Prompt NOx Formation Factors

Prompt NO_x formation is proportional to the number of carbon atoms present per unit volume and is independent of the parent hydrocarbon identity. The quantity of HCN formed increases with the concentration of hydrocarbon radicals, which in turn increases with equivalence ratio. As the equivalence ratio increases, prompt NO_x production increases at first, then passes a peak, and finally decreases due to a deficiency in oxygen.

9.1.4.4. Primary Reaction

The reaction described by Equation 9.16 (p. 369) is of primary importance. In recent studies [573] (p. 1086), comparison of probability density distributions for the location of the peak NO_x with those obtained for the peak CH have shown close correspondence, indicating that the majority of the NO_x at the flame base is prompt NO_x formed by the CH reaction. Assuming that the reaction described by Equation 9.16 (p. 369) controls the prompt NO_x formation rate,

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$\frac{d[NO]}{dt} = k_0[\mathrm{CH}][\mathrm{N_2}] \tag{9.22}$$

9.1.4.5. Modeling Strategy

There are, however, uncertainties about the rate data for the above reaction. From the reactions described by Equation 9.16 (p. 369) – Equation 9.20 (p. 369), it can be concluded that the prediction of prompt NO_x formation within the flame requires coupling of the NO_x kinetics to an actual hydrocarbon combustion mechanism. Hydrocarbon combustion mechanisms involve many steps and, as mentioned previously, are extremely complex and costly to compute. In the present NO_x model, a global kinetic parameter derived by De Soete [139] (p. 1061) is used. De Soete compared the experimental values of total NO_x formation rate with the rate of formation calculated by numerical integration of the empirical overall reaction rates of NO_x and N_2 formation. He showed that overall prompt formation rate can be predicted from the expression

$$\frac{d[NO]}{dt} = (\text{overall prompt NOx formation rate}) - (\text{overall prompt } N_2 \text{ formation rate}) \tag{9.23}$$

In the early stages of the flame, where prompt NO_x is formed under fuel-rich conditions, the O concentration is high and the N radical almost exclusively forms NO_x rather than nitrogen. Therefore, the prompt NO_x formation rate will be approximately equal to the overall prompt NO_x formation rate:

$$\frac{d[NO]}{dt} = k_{pr}[\mathrm{O_2}]^a[\mathrm{N_2}][\mathrm{FUEL}]e^{-E_a/RT} \tag{9.24}$$

For C_2H_4 (ethylene)-air flames,

$$k_{pr} = 1.2\times10^7(RT/p)^{a+1} \tag{9.25}$$

E_a is 251151 $\mathrm{J/mol}$, a is the oxygen reaction order, R is the universal gas constant, and p is pressure (all in SI units). The rate of prompt NO_x formation is found to be of the first order with respect to nitrogen and fuel concentration, but the oxygen reaction order, a, depends on experimental conditions.

9.1.4.6. Rate for Most Hydrocarbon Fuels

Equation 9.24 (p. 370) was tested against the experimental data obtained by Backmier et al. [35] (p. 1055) for different mixture and fuel types. The predicted results indicated that the model performance declined significantly under fuel-rich conditions and for higher hydrocarbon fuels. To reduce this error and predict the prompt NO_x adequately in all conditions, the De Soete model was modified using the available experimental data. A correction factor, f, was developed, which incorporates the effect of fuel type, that is, number of carbon atoms, and the air-to-fuel ratio for gaseous aliphatic hydrocarbons. Equation 9.24 (p. 370) now becomes

$$\frac{d[NO]}{dt} = f k'_{pr}[\mathrm{O_2}]^a[\mathrm{N_2}][\mathrm{FUEL}]e^{-E'_a/RT} \tag{9.26}$$

so that the source term due to prompt NO_x mechanism is

$$S_{prompt,NO} = M_{w,NO}\frac{d[NO]}{dt} \tag{9.27}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

In the above equations,

$$f = 4.75 + 0.0819n - 23.2\phi + 32\phi^2 - 12.2\phi^3 \tag{9.28}$$

$$k'_{pr} = 6.4 \times 10^6 (RT/p)^{a+1} \tag{9.29}$$

E'_a is 303474.125 $\mathrm{J/mol}$, n is the number of carbon atoms per molecule for the hydrocarbon fuel, and ϕ is the equivalence ratio. The correction factor is a curve fit for experimental data, valid for aliphatic alkane hydrocarbon fuels (C_nH_{2n+2}) and for equivalence ratios between 0.6 and 1.6. For values outside the range, the appropriate limit should be used. Values of k'_{pr} and E'_a were developed at the Department of Fuel and Energy at The University of Leeds in England.

Here, the concept of equivalence ratio refers to an overall equivalence ratio for the flame, rather than any spatially varying quantity in the flow domain. In complex geometries with multiple burners this may lead to some uncertainty in the specification of ϕ. However, since the contribution of prompt NO_x to the total NO_x emission is often very small, results are not likely to be significantly biased.

9.1.4.7. Oxygen Reaction Order

Oxygen reaction order depends on flame conditions. According to De Soete [139] (p. 1061), oxygen reaction order is uniquely related to oxygen mole fraction in the flame:

$$a = \begin{cases} 1.0, & X_{O_2} \le 4.1 \times 10^{-3} \\ -3.95 - 0.9 \ln X_{O_2}, & 4.1 \times 10^{-3} \le X_{O_2} \le 1.11 \times 10^{-2} \\ -0.35 - 0.1 \ln X_{O_2}, & 1.11 \times 10^{-2} < X_{O_2} < 0.03 \\ 0, & X_{O_2} \ge 0.03 \end{cases} \tag{9.30}$$

9.1.5. Fuel NOx Formation

9.1.5.1. Fuel-Bound Nitrogen

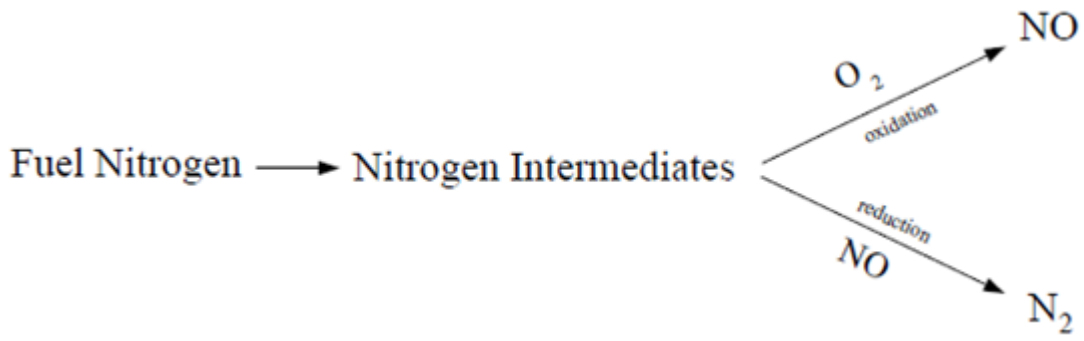
It is well known that nitrogen-containing organic compounds present in liquid or solid fossil fuel can contribute to the total NO_x formed during the combustion process. This fuel nitrogen is a particularly important source of nitrogen oxide emissions for residual fuel oil and coal, which typically contain 0.3–2% nitrogen by weight. Studies have shown that most of the nitrogen in heavy fuel oils is in the form of heterocycles, and it is thought that the nitrogen components of coal are similar [284] (p. 1069). It is believed that pyridine, quinoline, and amine type heterocyclic ring structures are of importance.

9.1.5.2. Reaction Pathways

The extent of the conversion of fuel nitrogen to NO_x is dependent on the local combustion characteristics and the initial concentration of nitrogen-bound compounds. Fuel-bound compounds that contain nitrogen are released into the gas phase when the fuel droplets or particles are heated during the devolatilization stage. From the thermal decomposition of these compounds (aniline, pyridine, pyrroles, and so on) in the reaction zone, radicals such as HCN, NH_3, N, CN, and NH can be formed and converted to NO_x. The above free radicals (that is, secondary intermediate nitrogen compounds) are subject to a double competitive reaction path. This chemical mechanism has been

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

subject to several detailed investigations [438] (p. 1078). Although the route leading to fuel NO_x formation and destruction is still not completely understood, different investigators seem to agree on a simplified model:



Recent investigations [256] (p. 1067) have shown that hydrogen cyanide appears to be the principal product if fuel nitrogen is present in aromatic or cyclic form. However, when fuel nitrogen is present in the form of aliphatic amines, ammonia becomes the principal product of fuel nitrogen conversion.

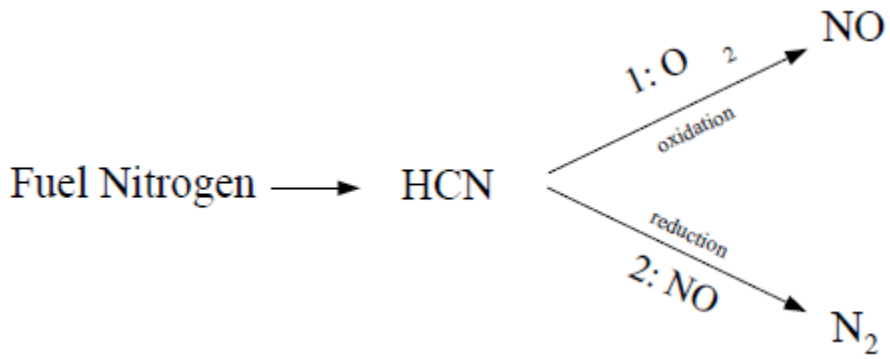
In the Ansys Fluent NO_x model, sources of NO_x emission for gaseous, liquid, and coal fuels are considered separately. The nitrogen-containing intermediates are grouped as HCN, NH_3, or a combination of both. Transport equations (Equation 9.1 (p. 364) and Equation 9.2 (p. 365) or Equation 9.3 (p. 365)) are solved, after which the source terms S_{HCN}, S_{NH_3}, and S_{NO} are determined for different fuel types. Discussions to follow refer to fuel NO_x sources for S_{NO} and intermediate HCN, and NH_3 sources for S_{HCN} and S_{NH_3}. Contributions from thermal and prompt mechanisms have been discussed in previous sections.

9.1.5.3. Fuel NOx from Gaseous and Liquid Fuels

The fuel NO_x mechanisms for gaseous and liquid fuels are based on different physics, but the same chemical reaction pathways.

9.1.5.3.1. Fuel NOx from Intermediate Hydrogen Cyanide (HCN)

When HCN is used as the intermediate species:



The source terms in the transport equations can be written as follows:

$$S_{HCN} = S_{pl,HCN} + S_{HCN-1} + S_{HCN-2} \tag{9.31}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$S_{NO} = S_{NO\text{–}1} + S_{NO\text{–}2} \tag{9.32}$$

9.1.5.3.1.1. HCN Production in a Gaseous Fuel

The rate of HCN production is equivalent to the rate of combustion of the fuel:

$$S_{pl,HCN} = \frac{\mathfrak{R}_{cf} Y_{N,fuel} M_{w,HCN}}{M_{w,N}} \tag{9.33}$$

where

$S_{pl,HCN}$ = source of HCN (kg/m^3–s)

$\mathfrak{R}_{cf}$ = mean limiting reaction rate of fuel (kg/m^3–s)

$Y_{N,fuel}$ = mass fraction of nitrogen in the fuel

The mean limiting reaction rate of fuel, $\mathfrak{R}_{cf}$, is calculated from the Magnussen combustion model, so the gaseous fuel NO_x option is available only when the generalized finite-rate model is used.

9.1.5.3.1.2. HCN Production in a Liquid Fuel

The rate of HCN production is equivalent to the rate of fuel release into the gas phase through droplet evaporation:

$$S_{pl,HCN} = \frac{S_{fuel} Y_{N,fuel} M_{w,HCN}}{M_{w,N} V} \tag{9.34}$$

where

$S_{pl,HCN}$ = source of HCN (kg/m^3–s)

S_{fuel} = rate of fuel release from the liquid droplets to the gas (kg/s)

$Y_{N,fuel}$ = mass fraction of nitrogen in the fuel

V = cell volume (m^3)

9.1.5.3.1.3. HCN Consumption

The HCN depletion rates from reactions (1) and (2) in the above mechanism are the same for both gaseous and liquid fuels, and are given by De Soete [139] (p. 1061) as

$$\mathfrak{R}_1 = A_1 X_{HCN} X_{O_2}^{a} e^{-E_1/RT} \tag{9.35}$$

$$\mathfrak{R}_2 = A_2 X_{HCN} X_{NO} e^{-E_2/RT} \tag{9.36}$$

where

$\mathfrak{R}_1, \mathfrak{R}_2$ = conversion rates of HCN (s^{-1})

T = instantaneous temperature (K)

X = mole fractions

A_1 = 1.0 $\times 10^{10}$ s^{-1}

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

A_2 = 3.0 $\times 10^{12}$ s^{-1}

E_1 = 280451.95 J/mol

E_2 = 251151 J/mol

The oxygen reaction order, a, is calculated from Equation 9.30 (p. 371).

Since mole fraction is related to mass fraction through molecular weights of the species ($M_{w,i}$) and the mixture ($M_{w,m}$),

$$X_i = Y_i \frac{M_{w,m}}{M_{w,i}} = \frac{Y_i}{M_{w,i}} \left(\frac{\rho RT}{p} \right) \tag{9.37}$$

9.1.5.3.1.4. HCN Sources in the Transport Equation

The mass consumption rates of HCN that appear in Equation 9.31 (p. 372) are calculated as

$$S_{HCN\text{–}1} = -\mathfrak{R}_1 \tag{9.38}$$

$$S_{HCN\text{–}2} = -\mathfrak{R}_2 \frac{M_{w,HCN} p}{R\overline{T}} \tag{9.39}$$

where

$S_{HCN\text{–}1}$, $S_{HCN\text{–}2}$ = consumption rates of HCN in reactions 1 and 2, respectively (kg/m^3–s)

p = pressure (Pa)

$\overline{T}$ = mean temperature (K)

R = universal gas constant

9.1.5.3.1.5. NOx Sources in the Transport Equation

NO_x is produced in reaction 1, but destroyed in reaction 2. The sources for Equation 9.32 (p. 373) are the same for a gaseous as for a liquid fuel, and are evaluated as follows:

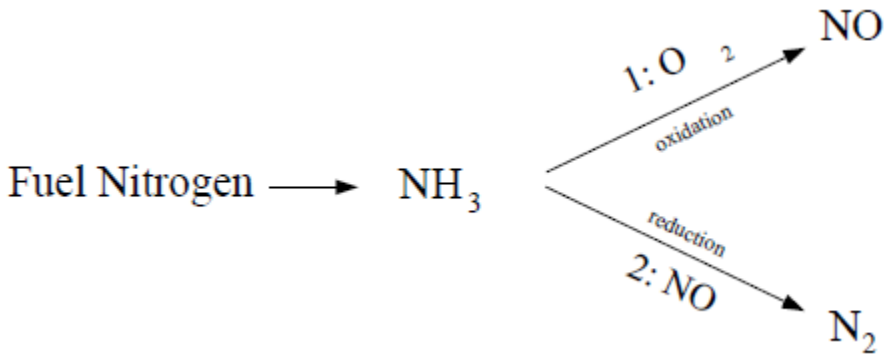
$$S_{NO\text{–}1} = -S_{HCN\text{–}1} \frac{M_{w,NO}}{M_{w,HCN}} = \mathfrak{R}_1 \frac{M_{w,NO} p}{R\overline{T}} \tag{9.40}$$

$$S_{NO\text{–}2} = S_{HCN\text{–}2} \frac{M_{w,NO}}{M_{w,HCN}} = -\mathfrak{R}_2 \frac{M_{w,NO} p}{R\overline{T}} \tag{9.41}$$

9.1.5.3.2. Fuel NOx from Intermediate Ammonia (NH3)

When NH_3 is used as the intermediate species:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.



The source terms in the transport equations can be written as follows:

$$S_{NH_3} = S_{pl,NH_3} + S_{NH_3\text{–}1} + S_{NH_3\text{–}2} \tag{9.42}$$

$$S_{NO} = S_{NO\text{–}1} + S_{NO\text{–}2} \tag{9.43}$$

9.1.5.3.2.1. NH3 Production in a Gaseous Fuel

The rate of NH_3 production is equivalent to the rate of combustion of the fuel:

$$S_{pl,NH_3} = \frac{\mathcal{R}_{cf} Y_{N,fuel} M_{w,NH_3}}{M_{w,N}} \tag{9.44}$$

where

S_{pl,NH_3} = source of NH_3 (kg/m^3–s)

$\mathcal{R}_{cf}$ = mean limiting reaction rate of fuel (kg/m^3–s)

$Y_{N,fuel}$ = mass fraction of nitrogen in the fuel

The mean limiting reaction rate of fuel, $\mathcal{R}_{cf}$, is calculated from the Magnussen combustion model, so the gaseous fuel NO_x option is available only when the generalized finite-rate model is used.

9.1.5.3.2.2. NH3 Production in a Liquid Fuel

The rate of NH_3 production is equivalent to the rate of fuel release into the gas phase through droplet evaporation:

$$S_{pl,NH_3} = \frac{S_{fuel} Y_{N,fuel} M_{w,NH_3}}{M_{w,N} V} \tag{9.45}$$

where

S_{pl,NH_3} = source of NH_3 (kg/m^3–s)

S_{fuel} = rate of fuel release from the liquid droplets to the gas (kg/s)

$Y_{N,fuel}$ = mass fraction of nitrogen in the fuel

V = cell volume (m^3)

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

9.1.5.3.2.3. NH3 Consumption

The NH_3 depletion rates from reactions (1) and (2) in the above mechanism are the same for both gaseous and liquid fuels, and are given by De Soete [139] (p. 1061) as

$$\begin{aligned} \mathfrak{R}_1 &= A_1 X_{NH_3} X_{O_2}^a e^{-E_1/RT} \\ \mathfrak{R}_2 &= A_2 X_{NH_3} X_{NO} e^{-E_2/RT} \end{aligned} \tag{9.46}$$

where

$\mathfrak{R}_1, \mathfrak{R}_2$ = conversion rates of NH_3 (s^{-1})

T = instantaneous temperature (K)

X = mole fractions

$A_1 = 4.0 \times 10^6$ s^{-1}

$A_2 = 1.8 \times 10^8$ s^{-1}

E_1 = 133947.2 J/mol

E_2 = 113017.95 J/mol

The oxygen reaction order, a, is calculated from Equation 9.30 (p. 371).

Since mole fraction is related to mass fraction through molecular weights of the species ($M_{w,i}$) and the mixture ($M_{w,m}$), X_i can be calculated using Equation 9.37 (p. 374).

9.1.5.3.2.4. NH3 Sources in the Transport Equation

The mass consumption rates of NH_3 that appear in Equation 9.42 (p. 375) are calculated as

$$S_{NH_3\text{-}1} = -\mathfrak{R}_1 \frac{M_{w,NH_3} p}{R\overline{T}} \tag{9.47}$$

$$S_{NH_3\text{-}2} = -\mathfrak{R}_2 \frac{M_{w,NH_3} p}{R\overline{T}} \tag{9.48}$$

where

$S_{NH_3\text{-}1}, S_{NH_3\text{-}2}$ = consumption rates of NH_3 in reactions 1 and 2, respectively (kg/m^3–s)

p = pressure (Pa)

$\overline{T}$ = mean temperature (K)

R = universal gas constant

9.1.5.3.2.5. NOx Sources in the Transport Equation

NO_x is produced in reaction 1, but destroyed in reaction 2. The sources for Equation 9.43 (p. 375) are the same for a gaseous as for a liquid fuel, and are evaluated as follows:

$$S_{NO\text{-}1} = -S_{NH_3\text{-}1} \frac{M_{w,NO}}{M_{w,NH_3}} = \mathfrak{R}_1 \frac{M_{w,NO} p}{R\overline{T}} \tag{9.49}$$

$$S_{NO\text{-}2} = S_{NH_3\text{-}2} \frac{M_{w,NO}}{M_{w,NH_3}} = -\mathfrak{R}_2 \frac{M_{w,NO} p}{R\overline{T}} \tag{9.50}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

9.1.5.3.3. Fuel NOx from Coal

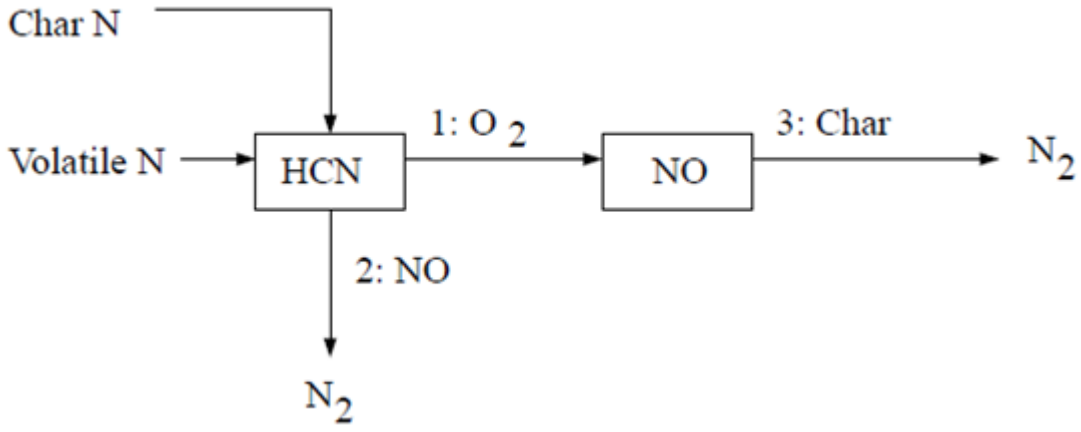
9.1.5.3.3.1. Nitrogen in Char and in Volatiles

For the coal it is assumed that fuel nitrogen is distributed between the volatiles and the char. Since there is no reason to assume that N is equally distributed between the volatiles and the char, the fraction of N in the volatiles and the char should be specified separately.

When HCN is used as the intermediate species, two variations of fuel NO_x mechanisms for coal are included. When NH_3 is used as the intermediate species, two variations of fuel NO_x mechanisms for coal are included, much like in the calculation of NO_x production from the coal via HCN. It is assumed that fuel nitrogen is distributed between the volatiles and the char.

9.1.5.3.3.2. Coal Fuel NOx Scheme A

The first HCN mechanism assumes that all char N converts to HCN, which is then converted partially to NO [610] (p. 1088). The reaction pathway is described as follows:



With the first scheme, all char-bound nitrogen converts to HCN. Thus,

$$S_{char,HCN} = \frac{S_c Y_{N,char} M_{w,HCN}}{M_{w,N} V} \tag{9.51}$$

$$S_{char,NO} = 0 \tag{9.52}$$

where

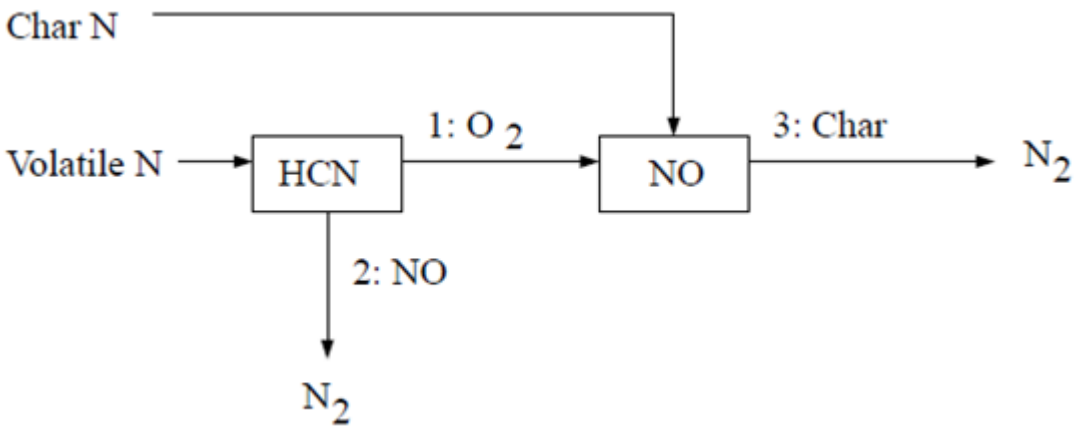
S_c = char burnout rate (kg/s)

$Y_{N,char}$ = mass fraction of nitrogen in char

V = cell volume (m^3)

9.1.5.3.3.3. Coal Fuel NOx Scheme B

The second HCN mechanism assumes that all char N converts to NO directly [384] (p. 1075). The reaction pathway is described as follows:



According to Lockwood [384] (p. 1075), the char nitrogen is released to the gas phase as NO directly, mainly as a desorption product from oxidized char nitrogen atoms. If this approach is followed, then

$$S_{char,HCN} = 0 \tag{9.53}$$

$$S_{char,NO} = \frac{S_c Y_{N,char} M_{w,NO}}{M_{w,N} V} \tag{9.54}$$

9.1.5.3.3.4. HCN Scheme Selection

The second HCN mechanism tends to produce more NO_x emission than the first. In general, however, it is difficult to say which one outperforms the other.

The source terms for the transport equations are

$$S_{HCN} = S_{pvc,HCN} + S_{HCN-1} + S_{HCN-2} \tag{9.55}$$

$$S_{NO} = S_{char,NO} + S_{NO-1} + S_{NO-2} + S_{NO-3} \tag{9.56}$$

The source contributions S_{HCN-1}, S_{HCN-2}, S_{NO-1}, and S_{NO-2} are described previously. Therefore, only the heterogeneous reaction source, S_{NO-3}, the char NO_x source, $S_{char,NO}$, and the HCN production source, $S_{pvc,HCN}$, need to be considered.

9.1.5.3.3.5. NOx Reduction on Char Surface

The heterogeneous reaction of NO reduction on the char surface has been modeled according to the following [358] (p. 1073):

$$\mathfrak{R}_3 = A_3 e^{-E_3/\overline{RT}} \overline{p}_{NO} \tag{9.57}$$

where

$\mathfrak{R}_3$ = rate of NO reduction $\left(\mathrm{mol}/\mathrm{m}^2_{\mathrm{BET}}\text{-s}\right)$

$\overline{p}_{NO}$ = mean NO partial pressure (atm)

E_3 = 142737.485J/mol

A_3 = 230 $\mathrm{mol}/\mathrm{m}^2_{\mathrm{BET}}$-s-atm

$\overline{T}$ = mean temperature (K)

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

The partial pressure $\overline{p}_{NO}$ is calculated using Dalton's law:

$$\overline{p}_{NO}=\overline{p}X_{NO}$$

The rate of NO consumption due to reaction 3 will then be

$$S_{NO\text{–}3}=c_s A_{BET} M_{w,NO} \mathfrak{R}_3$$

where

A_{BET} = BET surface area (m^2/kg)

c_s = concentration of particles (kg/m^3)

$S_{NO\text{–}3}$ = NO consumption (kg/m^3-s)

9.1.5.3.3.5.1. BET Surface Area

The heterogeneous reaction involving char is mainly an adsorption process, whose rate is directly proportional to the pore surface area. The pore surface area is also known as the BET surface area, due to the researchers who pioneered the adsorption theory (Brunauer, Emmett, and Teller [85] (p. 1057)). For commercial adsorbents, the pore (BET) surface areas range from 100,000 to 2 million square meters per kilogram, depending on the microscopic structure. For coal, the BET area is typically 25,000 m^2/k g, which is used as the default in Ansys Fluent. The overall source of HCN ($S_{pvc,HCN}$) is a combination of volatile contribution ($S_{vol,HCN}$) and char contribution ($S_{char,HCN}$):

$$S_{pvc,HCN}=S_{vol,HCN}+S_{char,HCN}$$

9.1.5.3.3.5.2. HCN from Volatiles

The source of HCN from the volatiles is related to the rate of volatile release:

$$S_{vol,HCN}=\frac{S_{vol}Y_{N,vol}M_{w,HCN}}{M_{w,N}V}$$

where

S_{vol} = source of volatiles originating from the coal particles into the gas phase (kg/s)

$Y_{N,vol}$ = mass fraction of nitrogen in the volatiles

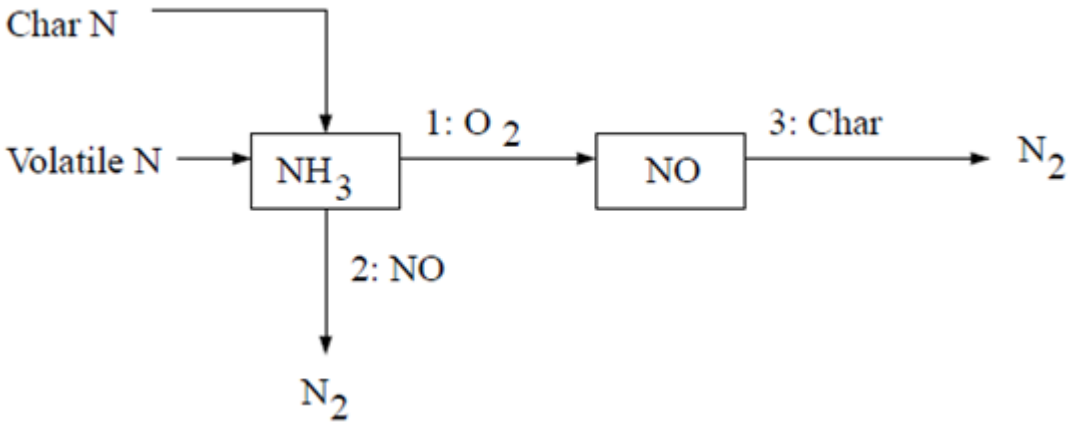
V = cell volume (m^3)

Calculation of sources related to char-bound nitrogen depends on the fuel NO_x scheme selection.

9.1.5.3.3.6. Coal Fuel NOx Scheme C

The first NH_3 mechanism assumes that all char N converts to NH_3 which is then converted partially to NO [610] (p. 1088). The reaction pathway is described as follows:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.



In this scheme, all char-bound nitrogen converts to NH_3. Thus,

$$S_{char,NH_3} = \frac{S_c Y_{N,char} M_{w,NH_3}}{M_{w,N} V} \tag{9.58}$$

$$S_{char,NO} = 0 \tag{9.59}$$

where

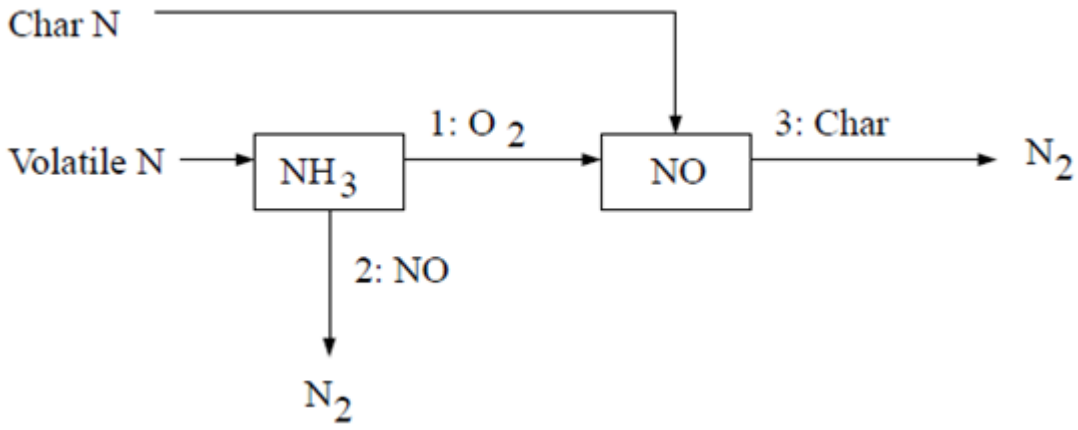
S_c = char burnout rate (kg/s)

$Y_{N,char}$ = mass fraction of nitrogen in char

V = cell volume (m^3)

9.1.5.3.3.7. Coal Fuel NOx Scheme D

The second NH_3 mechanism assumes that all char N converts to NO directly [384] (p. 1075). The reaction pathway is described as follows:



According to Lockwood [384] (p. 1075), the char nitrogen is released to the gas phase as NO directly, mainly as a desorption product from oxidized char nitrogen atoms. If this approach is followed, then

$$S_{char,NH_3} = 0 \tag{9.60}$$

$$S_{char,NO} = \frac{S_c Y_{N,char} M_{w,NO}}{M_{w,N} V} \tag{9.61}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

9.1.5.3.3.8. NH3 Scheme Selection

The second NH_3 mechanism tends to produce more NO_x emission than the first. In general, however, it is difficult to say which one outperforms the other.

The source terms for the transport equations are

$$S_{NH_3} = S_{pvc,NH_3}+S_{NH_3\text{–}1}+S_{NH_3\text{–}2} \tag{9.62}$$

$$S_{NO} = S_{char,NO}+S_{NO\text{–}1}+S_{NO\text{–}2}+S_{NO\text{–}3} \tag{9.63}$$

The source contributions $S_{NH_3\text{–}1}$, $S_{NH_3\text{–}2}$, $S_{NO\text{–}1}$, $S_{NO\text{–}2}$, $S_{NO\text{–}3}$, and $S_{char,NO}$ are described previously. Therefore, only the NH_3 production source, S_{pvc,NH_3}, must be considered.

The overall production source of NH_3 is a combination of volatile contribution (S_{vol,NH_3}), and char contribution (S_{char,NH_3}):

$$S_{pvc,NH_3}=S_{vol,NH_3}+S_{char,NH_3} \tag{9.64}$$

9.1.5.3.3.8.1. NH3 from Volatiles

The source of NH_3 from the volatiles is related to the rate of volatile release:

$$S_{vol,NH_3}=\frac{S_{vol}Y_{N,vol}M_{w,NH_3}}{M_{w,N}V}$$

where

S_{vol} = source of volatiles originating from the coal particles into the gas phase (kg/s)

$Y_{N,vol}$ = mass fraction of nitrogen in the volatiles

V = cell volume (m^3)

Calculation of sources related to char-bound nitrogen depends on the fuel NO_x scheme selection.

9.1.5.3.4. Fuel Nitrogen Partitioning for HCN and NH3 Intermediates

In certain cases, especially when the fuel is a solid, both HCN and NH_3 can be generated as intermediates at high enough temperatures [469] (p. 1080). In particular, low-ranking (lignite) coal has been shown to produce 10 times more NH_3 compared to the level of HCN, whereas higher-ranking (bituminous) coal has been shown to produce only HCN [468] (p. 1080). Studies by Winter et al. [712] (p. 1094) have shown that for bituminous coal, using an HCN/NH_3 partition ratio of 9:1 gave better NO_x predictions when compared to measurements than specifying only a single intermediate species. Liu and Gibbs [383] (p. 1075) work with woody-biomass (pine wood chips), on the other hand, has suggested an HCN/NH_3 ratio of 1:9 due to the younger age of the fuel.

In total, the above work suggests the importance of being able to specify that portions of the fuel nitrogen will be converted to both HCN and NH_3 intermediates at the same time. In Ansys Fluent, fuel nitrogen partitioning can be used whenever HCN or NH_3 are intermediates for NO_x production, though it is mainly applicable to solid fuels such as coal and biomass. The reaction pathways and source terms for HCN and NH_3 are described in previous sections.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

9.1.6. NOx Formation from Intermediate N2O

Melte and Pratt [423] (p. 1077) proposed the first intermediate mechanism for NO_x formation from molecular nitrogen (N_2) via nitrous oxide (N_2O). Nitrogen enters combustion systems mainly as a component of the combustion and dilution air. Under favorable conditions, which are elevated pressures and oxygen-rich conditions, this intermediate mechanism can contribute as much as 90% of the NO_x formed during combustion. This makes it particularly important in industrial application such as gas turbines and compression-ignition engines. Because these devices are operated at increasingly low temperatures to prevent NO_x formation via the thermal NO_x mechanism, the relative importance of the N_2O-intermediate mechanism is increasing. It has been observed that about 30% of the NO_x formed in these systems is mainly related to the N_2O-intermediate mechanism.

The N_2O-intermediate mechanism may also be of importance in systems operated in flameless mode (for example, diluted combustion, flameless combustion, flameless oxidation, and FLOX systems). In a flameless mode, fuel and oxygen are highly diluted in inert gases so that the combustion reactions and resulting heat release are carried out in the diffuse zone. As a consequence, elevated peaks of temperature are avoided, which prevents thermal NO_x. Research suggests that the N_2O-intermediate mechanism may contribute about 90% of the NO_x formed in flameless mode, and that the remainder can be attributed to the prompt NO_x mechanism. The relevance of NO_x formation from N_2O has been observed indirectly and theoretically speculated for a number of combustion systems, by a number of researchers [40] (p. 1055), [122] (p. 1060), [207] (p. 1064), [629] (p. 1089), [638] (p. 1090).

9.1.6.1. N2O - Intermediate NOx Mechanism

The simplest form of the mechanism [423] (p. 1077) takes into account two reversible elementary reactions:

$$\mathrm{N_2+O+M} \rightleftharpoons \mathrm{N_2O+M} \tag{9.65}$$

$$\mathrm{N_2O+O} \rightleftharpoons \mathrm{2NO} \tag{9.66}$$

Here, M is a general third body. Because the first reaction involves third bodies, the mechanism is favored at elevated pressures. Both reactions involve the oxygen radical O, which makes the mechanism favored for oxygen-rich conditions. While not always justified, it is often assumed that the radical O atoms originate solely from the dissociation of molecular oxygen,

$$\frac{1}{2}\mathrm{O_2} \rightleftharpoons \mathrm{O} \tag{9.67}$$

According to the kinetic rate laws, the rate of NO_x formation via the N_2O-intermediate mechanism is

$$\frac{d[NO]}{dt} = 2\left(k_{f,2}[N_2O][O] - k_{r,2}[NO]^2\right) \qquad \mathrm{mol/m^3\text{-}s} \tag{9.68}$$

To solve Equation 9.68 (p. 382), you will need to have first calculated [O] and [N_2O].

It is often assumed that N_2O is at quasi-steady-state (that is, $d[\mathrm{N_2O}]/dt = 0$), which implies

$$[\mathrm{N_2O}] = \frac{k_{f,1}[\mathrm{N_2}][\mathrm{O}][\mathrm{M}] + k_{r,2}[\mathrm{NO}]^2}{k_{r,1}[\mathrm{M}] + k_{f,2}[\mathrm{O}]} \tag{9.69}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

The system of Equation 9.68 (p. 382) – Equation 9.69 (p. 382) can be solved for the rate of NO_x formation when the concentration of N_2, O_2, and M, the kinetic rate constants for Equation 9.65 (p. 382) and Equation 9.66 (p. 382), and the equilibrium constant of Equation 9.67 (p. 382) are known. The appearance of NO in Equation 9.66 (p. 382) entails that coupling of the N_2O mechanism with the thermal NO_x mechanism (and other NO_x mechanisms).

$$k_{f,1} = 4.44\times10^{32}T^{-8.358}e^{-28234/T} \qquad k_{r,1} = 4.00\times10^{8}e^{-28234/T}$$

$$k_{f,2} = 2.90\times10^{7}e^{-11651/T} \qquad k_{r,2} = 1.45\times10^{-29}T^{9.259}e^{-11651/T}$$

In the above expressions, $k_{f,1}$ and $k_{f,2}$ are the forward rate constants of Equation 9.65 (p. 382) and Equation 9.66 (p. 382), and $k_{r,1}$ and $k_{r,2}$ are the corresponding reverse rate constants. The units for $k_{f,2}$, $k_{r,1}$, and $k_{r,2}$ are m^3/mol-s, while $k_{f,1}$ has units of m^6/mol^2-s.

9.1.7. NOx Reduction by Reburning

The design of complex combustion systems for utility boilers, based on air- and fuel-staging technologies, involves many parameters and their mutual interdependence. These parameters include the local stoichiometry, temperature and chemical concentration field, residence time distribution, velocity field, and mixing pattern. A successful application of the in-furnace reduction techniques requires control of these parameters in an optimum manner, so as to avoid impairing the boiler performance. In the mid 1990s, global models describing the kinetics of NO_x destruction in the reburn zone of a staged combustion system became available. Two of these models are described below.

9.1.7.1. Instantaneous Approach

The instantaneous NO_x reburning mechanism is a pathway whereby NO reacts with hydrocarbons and is subsequently reduced. In general:

$$\mathrm{CH}_i+\mathrm{NO}\rightarrow\mathrm{HCN}+\mathrm{products} \tag{9.70}$$

Three reburn reactions are modeled by Ansys Fluent for $1600\leq T\leq2100$:

$$\mathrm{CH}+\mathrm{NO} \xrightarrow{k_1} \mathrm{HCN}+\mathrm{O} \tag{9.71}$$

$$\mathrm{CH}_2+\mathrm{NO} \xrightarrow{k_2} \mathrm{HCN}+\mathrm{OH} \tag{9.72}$$

$$\mathrm{CH}_3+\mathrm{NO} \xrightarrow{k_3} \mathrm{HCN}+\mathrm{H}_2\mathrm{O} \tag{9.73}$$

Important:

If the temperature is outside of this range, NO reburn will not be computed.

The rate constants for these reactions are taken from Bowman [72] (p. 1057) and have units of m^3/mol-s:

$$k_1=1\times10^{8};\ \ k_2=1.4\times10^{6}e^{-550/T};\ \ k_3=2\times10^{5}$$

The NO depletion rate due to reburn is expressed as

$$\frac{d[\mathrm{NO}]}{dt} = -k_1[\mathrm{CH}][\mathrm{NO}] - k_2[\mathrm{CH_2}][\mathrm{NO}] - k_3[\mathrm{CH_3}][\mathrm{NO}] \tag{9.74}$$

and the source term for the reburning mechanism in the NO transport equation can be calculated as

$$S_{reburn,NO} = -M_{w,NO}\frac{d[\mathrm{NO}]}{dt} \tag{9.75}$$

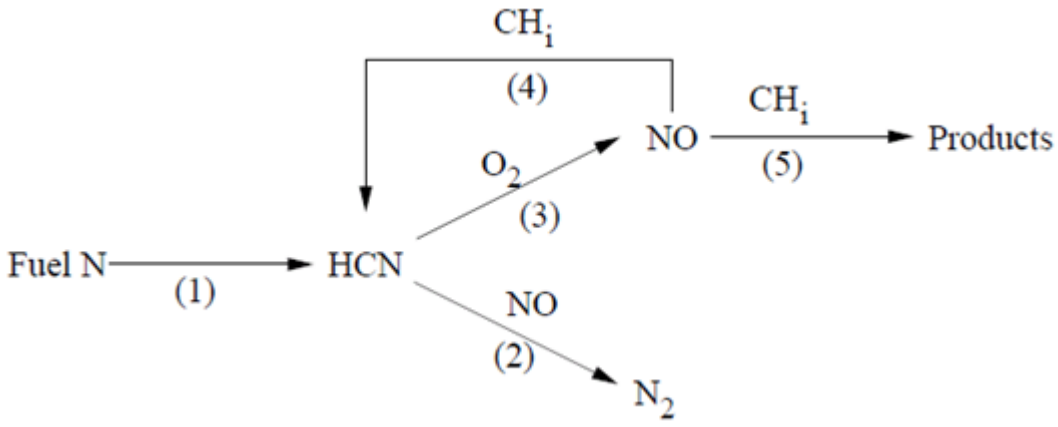
Important:

To calculate the NO depletion rate due to reburning, Ansys Fluent will obtain the concentrations of CH, CH_2, and CH_3 from the species mass fraction results of the combustion calculation. When you use this method, you must be sure to include the species CH, CH_2, and CH_3 in your problem definition.

9.1.7.2. Partial Equilibrium Approach

The partial equilibrium approach is based on the model proposed by Kandamby et al. [288] (p. 1069), [21] (p. 1054). The model adds a reduction path to De Soete's global model [139] (p. 1061) that describes the NO_x formation/destruction mechanism in a pulverized coal flame. The additional reduction path accounts for the NO_x destruction in the fuel-rich reburn zone by CH radicals (see Figure 9.1: De Soete's Global NOx Mechanism with Additional Reduction Path (p. 384)).

Figure 9.1: De Soete's Global NOx Mechanism with Additional Reduction Path



This model can be used in conjunction with the eddy-dissipation combustion model and does not require the specification of CH radical concentrations, because they are computed based on the CH-radical partial equilibrium. The reburn fuel itself can be an equivalent of CH_4, CH_3, CH_2, or CH. How this equivalent fuel is determined is open for debate, and an approximate guide would be to consider the C/H ratio of the fuel itself. A multiplicative constant of 4.0×10^{-4} has been developed for the partial equilibrium of CH radicals to reduce the rates of HCN and NO in the reburn model. This value was obtained by researchers who developed the model, by way of predicting NO_x values for a number of test cases for which experimental data exists.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

9.1.7.2.1. NOx Reduction Mechanism

In the fuel-rich reburn zone, the HCN oxidation is suppressed and the amount of NO formed in the primary combustion zone is decreased by the reduction reaction from HCN to N_2. However, the NO concentration may also decrease due to reactions with CH radicals, which are available in significant amounts in the reburn zone. The following are considered to be the most important reactions of NO reduction by CH radicals:

$$NO+CH_2 \rightarrow HCN+OH \tag{9.76}$$

$$NO+CH \rightarrow HCN+O \tag{9.77}$$

$$NO+C \rightarrow CN+O \tag{9.78}$$

These reactions may be globally described by the addition of pathways (4) and (5) in Figure 9.1: De Soete's Global NOx Mechanism with Additional Reduction Path (p. 384), leading respectively to the formation of HCN and of minor intermediate nitrogen radicals. Assuming that methane is the reburning gas, the global NO reduction rates are then expressed as

$$\mathcal{R}_4 = \left(k_a\chi_1+k_b\chi_1^2\right)[CH_4][NO] \tag{9.79}$$

$$\mathcal{R}_5 = k_c\chi_1^3\chi_2[CH_4][NO] \tag{9.80}$$

where

$$\chi_1=\frac{[H]}{[H_2]};\chi_2=\frac{[OH]}{[H_2O]}$$

Therefore, the additional source terms of the HCN and NO transport equations due to reburn reactions are given by

$$\frac{d[HCN]}{dt} = 4\times10^{-4}\mathcal{R}_4 \tag{9.81}$$

$$\frac{d[NO]}{dt} = -4\times10^{-4}\left(\mathcal{R}_4+\mathcal{R}_5\right) \tag{9.82}$$

Certain assumptions are required to evaluate the rate constants k_a, k_b, and k_c and the factors χ_1 and χ_2. For hydrocarbon diffusion flames, the following reaction set can be reasonably considered to be in partial equilibrium:

$$CH_4+H \rightleftharpoons CH_3+H_2 \tag{9.83}$$

$$CH_3+OH \rightleftharpoons CH_2+H_2O \tag{9.84}$$

$$CH_2+H \rightleftharpoons CH+H_2 \tag{9.85}$$

$$CH+H \rightleftharpoons C+H_2 \tag{9.86}$$

Thus, the rate constants may be computed as

$$k_a=k_1\frac{k_{f,4}k_{f,5}}{k_{r,4}k_{r,5}};\ k_b=k_2\frac{k_{f,4}k_{f,5}k_{f,6}}{k_{r,4}k_{r,5}k_{r,6}};\ k_c=k_3\frac{k_{f,4}k_{f,5}k_{f,6}k_{f,7}}{k_{r,4}k_{r,5}k_{r,6}k_{r,7}}$$

where k_1, k_2, and k_3 are the rate constants for Equation 9.76 (p. 385) – Equation 9.78 (p. 385). The forward and reverse rate constants for Equation 9.83 (p. 385) – Equation 9.86 (p. 385) are $k_{f,4}-k_{f,7}$ and $k_{r,4}-k_{r,7}$, respectively. In addition, it is assumed that $\chi_1=1$, because the H-radical concentration

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

in the post-flame region of a hydrocarbon diffusion flame has been observed to be of the same order as [H_2]. Finally, the OH-radical concentration is estimated by considering the reaction

$$OH+H_2 \rightleftharpoons H_2O+H \tag{9.87}$$

to be partially equilibrated, leading to the relationship

$$\chi_2=\frac{k_{r,8}}{k_{f,8}}$$

Values for the rate constants k_a, k_b, and k_c for different equivalent fuel types are given in Arrhenius form ($AT^b e^{-E/RT}$) in Table 9.1: Rate Constants for Different Reburn Fuels (p. 386) [355] (p. 1073). All rate constants have units of m^3/mol-s, and all values of E have units of J/mol.

Table 9.1: Rate Constants for Different Reburn Fuels

Equivalent Fuel Type	k_a			k_b			k_c		
	A	b	E	A	b	E	A	b	E
CH_4	5.30×10^{9}	-1.54	27977	3.31×10^{13}	-3.33	15090	3.06×10^{11}	-2.64	77077
CH_3	0.37×10^{9}	-1.54	27977	0.23×10^{13}	-3.33	15090	0.21×10^{11}	-2.64	77077
CH_2	0.23×10^{7}	-1.54	27977	0.14×10^{11}	-3.33	15090	0.13×10^{9}	-2.64	77077
CH	0.0	0.0	0.0	0.63×10^{8}	-3.33	15090	0.58×10^{6}	-2.64	77077

For Equation 9.87 (p. 386),

$$k_{f,8}=1.02\times10^{5}T^{1.60}e^{-13802/RT};\quad k_{r,8}=4.52\times10^{5}T^{1.60}e^{-80815/RT}$$

9.1.8. NOx Reduction by SNCR

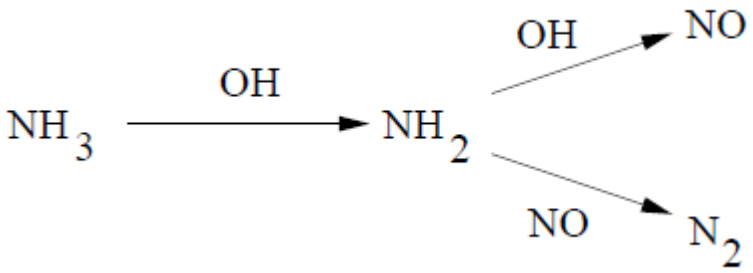
The selective noncatalytic reduction of NO_x (SNCR), first described by Lyon [400] (p. 1076), is a method to reduce the emission of NO_x from combustion by injecting a selective reductant such as ammonia (NH_3) or urea ($CO(NH_2)_2$) into the furnace, where it can react with NO in the flue gas to form N_2. However, the reductant can be oxidized as well to form NO_x. The selectivity for the reductive reactions decreases with increasing temperature [436] (p. 1078), while the rate of the initiation reaction simultaneously increases. This limits the SNCR process to a narrow temperature interval, or window, where the lower temperature limit for the interval is determined by the residence time.

9.1.8.1. Ammonia Injection

Several investigators have modeled the process using a large number of elementary reactions. A simple empirical model has been proposed by Fenimore [172] (p. 1062), which is based on experimental measurements. However, the model was found to be unsuitable for practical applications. Ostberg and Dam-Johansen [498] (p. 1082) proposed a two-step scheme describing the SNCR process as shown in Figure 9.2: Simplified Reaction Mechanism for the SNCR Process (p. 387), which is a single initiation step followed by two parallel reaction pathways: one leading to NO reduction, and the other to NO formation.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Figure 9.2: Simplified Reaction Mechanism for the SNCR Process



$$NO+NH_3+\frac{1}{4}O_2 \rightarrow N_2+\frac{3}{2}H_2O \quad (9.88)$$

$$NH_3+\frac{5}{4}O_2 \rightarrow NO+\frac{3}{2}H_2O \quad (9.89)$$

The reaction orders of NO and NH_3 at 4% volume O_2 and the empirical rate constants k_r and k_{ox} for Equation 9.88 (p. 387) Equation 9.89 (p. 387), respectively, have been estimated from work done by Brouwer et al. [82] (p. 1057). The reaction order of NO was found to be 1 for Equation 9.88 (p. 387) and the order of NH_3 was also found to be 1 for both reactions. As such, the following reaction rates for NO and NH_3, at 4% volume O_2, were proposed:

$$\mathfrak{R}_{NO} = -k_r[NO][NH_3]+k_{ox}[NH_3][O_2] \quad (9.90)$$

$$\mathfrak{R}_{NH_3} = -k_r[NO][NH_3]-k_{ox}[NH_3][O_2] \quad (9.91)$$

The rate constants k_r and k_{ox} have units of m^3/mol-s, and are defined as

$$k_r=4.24\times10^2T^{5.30}e^{-E_r/RT};\ \ k_{ox}=3.50\times10^{-1}T^{7.65}e^{-E_{ox}/RT}$$

where E_r=349937.06 J/mol and E_{ox}=524487.005 J/mol.

This model has been shown to give reasonable predictions of the SNCR process in pulverized coal and fluidized bed combustion applications. The model also captures the influence of the most significant parameters for SNCR, which are the temperature of the flue gas at the injection position, the residence time in the relevant temperature interval, the NH_3 to NO molar ratio, and the effect of combustible additives. This model overestimates the NO reduction for temperatures above the optimum temperature by an amount similar to that of the detailed kinetic model of Miller and Bowman [436] (p. 1078).

Important:

The SNCR process naturally occurs when NH_3 is present in the flame as a fuel N intermediate. For this reason, even if the SNCR model is not activated and there is no reagent injection, the natural SNCR process may still occur in the flame. The temperature range or "window" at which SNCR may occur is 1073 K < T < 1373 K. If you want to model your case without using the natural SNCR process, contact your support engineer for information on how to deactivate it.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

9.1.8.2. Urea Injection

Urea as a reagent for the SNCR process is similar to that of injecting ammonia, and has been used in power station combustors to reduce NO emissions successfully. However, both reagents, ammonia and urea, have major limitations as a NO_x reducing agent. The narrow temperature "window" of effectiveness and mixing limitations are difficult factors to handle in a large combustor. The use of urea instead of ammonia as the reducing agent is attractive because of the ease of storage and handling of the reagent.

The SNCR process using urea is a combination of Thermal DeNOx (SNCR with ammonia) and RAPRENOx (SNCR using cyanuric acid that, under heating, sublimes and decomposes into isocyanic acid), because urea most probably decomposes into ammonia and isocyanic acid [436] (p. 1078).

One problem of SNCR processes using urea is that slow decay of HNCO, as well as the reaction channels leading to N_2O and CO, can significantly increase the emission of pollutants other than NO. Urea seems to involve a significant emission of carbon-containing pollutants, such as CO and HNCO.

Also, some experimental observations [557] (p. 1085) show that SNCR using urea is effective in a narrow temperature window that is shifted toward higher temperatures, when compared to Thermal DeNOx processes at the same value of the ratio of nitrogen in the reducing agent and the NO in the feed, β, where β is defined as the ratio of nitrogen in the reducing agent and NO in the feed. The effect of increasing the β value is to increase the efficiency of abatement, while the effect of increasing O_2 concentration depends on the temperature.

The model described here is proposed by Brouwer et al. [82] (p. 1057) and is a seven-step reduced kinetic mechanism. Brouwer et al. [82] (p. 1057) assumes that the breakdown of urea is instantaneous and 1 mole of urea is assumed to produce 1.1 moles of NH_3 and 0.9 moles of HNCO. The work of Rota et al. [557] (p. 1085) proposed a finite rate two-step mechanism for the breakdown of urea into ammonia and HNCO.

The seven-step reduced mechanism is given in Table 9.2: Seven-Step Reduced Mechanism for SNCR with Urea (p. 388), and the two-step urea breakdown mechanism is given in Table 9.3: Two-Step Urea Breakdown Process (p. 389).

Table 9.2: Seven-Step Reduced Mechanism for SNCR with Urea

Reaction	A	b	E
$NH_3 + NO \rightarrow N_2 + H_2O + H$	4.24E+02	5.30	349937.06
$NH_3 + O_2 \rightarrow NO + H_2O + H$	3.500E-01	7.65	524487.005
$HNCO + M \rightarrow H + NCO + M$	2.400E+08	0.85	284637.8
$NCO + NO \rightarrow N_2O + CO$	1.000E+07	0.00	-1632.4815
$NCO + OH \rightarrow NO + CO + H$	1.000E+07	0.00	0
$N_2O + OH \rightarrow N_2 + O_2 + H$	2.000E+06	0.00	41858.5

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Reaction	A	b	E
$N_2O + M \rightarrow N_2 + O + M$	6.900E+17	-2.5	271075.646

Table 9.3: Two-Step Urea Breakdown Process

Reaction	A	b	E
$CO(NH_2)_2 \rightarrow NH_3 + HNCO$	1.27E+04	0	65048.109
$CO(NH_2)_2 + H_2O \rightarrow 2NH_3 + CO_2$	6.13E+04	0	87819.133

where SI units (m, mol, sec, J) are used in Table 9.2: Seven-Step Reduced Mechanism for SNCR with Urea (p. 388) and Table 9.3: Two-Step Urea Breakdown Process (p. 389).

9.1.8.3. Transport Equations for Urea, HNCO, and NCO

When the SNCR model with urea injection is employed in addition to the usual transport equations, Ansys Fluent solves the following three additional mass transport equations for the urea, HNCO, and NCO species.

$$\frac{\partial}{\partial t}\left(\rho Y_{CO(NH_2)_2}\right) + \nabla \cdot \left(\rho \vec{v} Y_{CO(NH_2)_2}\right) = \nabla \cdot \left(\rho \mathcal{D} Y_{CO(NH_2)_2}\right) + S_{CO(NH_2)_2} \tag{9.92}$$

$$\frac{\partial}{\partial t}\left(\rho Y_{HNCO}\right) + \nabla \cdot \left(\rho \vec{v} Y_{HNCO}\right) = \nabla \cdot \left(\rho \mathcal{D} Y_{HNCO}\right) + S_{HNCO} \tag{9.93}$$

$$\frac{\partial}{\partial t}\left(\rho Y_{NCO}\right) + \nabla \cdot \left(\rho \vec{v} Y_{NCO}\right) = \nabla \cdot \left(\rho \mathcal{D} Y_{NCO}\right) + S_{NCO} \tag{9.94}$$

where $Y_{CO(NH_2)_2}$, Y_{HNCO} and Y_{NCO} are mass fractions of urea, HNCO, and NCO in the gas phase. The source terms $S_{CO(NH_2)_2}$, S_{HNCO}, and S_{NCO} are determined according to the rate equations given in Table 9.2: Seven-Step Reduced Mechanism for SNCR with Urea (p. 388) and Table 9.3: Two-Step Urea Breakdown Process (p. 389) and the additional source terms due to reagent injection. These additional source terms are determined next. The source terms in the transport equations can be written as follows:

$$S_{CO(NH_2)_2} = S_{pl,CO(NH_2)_2} + S_{CO(NH_2)_2-reac} \tag{9.95}$$

$$S_{HNCO} = S_{pl,HNCO} + S_{HNCO-reac} \tag{9.96}$$

$$S_{NCO} = S_{NCO-reac} \tag{9.97}$$

Apart from the source terms for the above three species, additional source terms for NO, NH_3, and N_2O are also determined as follows, which should be added to the previously calculated sources due to fuel NO_x:

$$S_{NO} = S_{NO-reac} \tag{9.98}$$

$$S_{NH_3} = S_{pl,NH_3} + S_{NH_3-reac} \tag{9.99}$$

$$S_{N_2O} = S_{N_2O-reac} \tag{9.100}$$

The source terms S_{i-reac} for i^{th} species are determined from the rate equations given in Table 9.2: Seven-Step Reduced Mechanism for SNCR with Urea (p. 388) and Table 9.3: Two-Step Urea Breakdown Process (p. 389).

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

9.1.8.4. Urea Production due to Reagent Injection

The rate of urea production is equivalent to the rate of reagent release into the gas phase through droplet evaporation:

$$S_{pl,CO(NH_2)_2} = \frac{S_{reagent}}{V} \tag{9.101}$$

where $S_{reagent}$ is the rate of reagent release from the liquid droplets to the gas phase (kg/s) and V is the cell volume (m^3).

9.1.8.5. NH3 Production due to Reagent Injection

If the urea decomposition model is set to the **user-specified** option, then the rate of NH_3 production is proportional to the rate of reagent release into the gas phase through droplet evaporation:

$$S_{pl,NH_3} = MCF_{NH_3} \frac{S_{reagent}}{V} \tag{9.102}$$

where $S_{reagent}$ is the rate of reagent release from the liquid droplets to the gas phase (kg/s), MCF_{NH_3} is the mole fraction of NH_3 in the NH_3/ HNCO mixture created from urea decomposition, and V is the cell volume (m^3).

9.1.8.6. HNCO Production due to Reagent Injection

If the urea decomposition model is set to the **user-specified** option, then the rate of HNCO production is proportional to the rate of reagent release into the gas phase through droplet evaporation:

$$S_{pl,HNCO} = MCF_{HNCO} \frac{S_{reagent}}{V} \tag{9.103}$$

where $S_{reagent}$, the injection source term, is the rate of reagent release from the liquid droplets to the gas phase (kg/s), MCF_{HNCO} is the mole fraction of HNCO in the NH_3/ HNCO mixture created from urea decomposition, and V is the cell volume (m^3).

Important:

The mole conversion fractions (MCF) for species NH_3 and HNCO are determined through the user species values such that if one mole of urea decomposes into 1.1 moles of NH_3 and 0.9 moles of HNCO, then MCF_{NH_3} = 0.55 and MCF_{HNCO} = 0.45. When the **user-specified** option is used for urea decomposition, then $S_{pl,CO(NH_2)_2} = 0$.

However, the default option for urea decomposition is through rate limiting reactions given in Table 9.3: Two-Step Urea Breakdown Process (p. 389), and the source terms are calculated accordingly. In this case, both values of S_{pl,NH_3} and $S_{pl,HNCO}$ are zero.

9.1.9. NOx Formation in Turbulent Flows

The kinetic mechanisms of NO_x formation and destruction described in the preceding sections have all been obtained from laboratory experiments using either a laminar premixed flame or shock-tube

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

studies where molecular diffusion conditions are well defined. In any practical combustion system, however, the flow is highly turbulent. The turbulent mixing process results in temporal fluctuations in temperature and species concentration that will influence the characteristics of the flame.

The relationships among NO_x formation rate, temperature, and species concentration are highly nonlinear. Hence, if time-averaged composition and temperature are employed in any model to predict the mean NO_x formation rate, significant errors will occur. Temperature and composition fluctuations must be taken into account by considering the probability density functions that describe the time variation.

9.1.9.1. The Turbulence-Chemistry Interaction Model

In turbulent combustion calculations, Ansys Fluent solves the density-weighted time-averaged Navier-Stokes equations for temperature, velocity, and species concentrations or mean mixture fraction and variance. To calculate NO concentration, a time-averaged NO formation rate must be computed at each point in the domain using the averaged flow-field information.

Methods of modeling the mean turbulent reaction rate can be based on either moment methods [711] (p. 1094) or probability density function (PDF) techniques [278] (p. 1069). Ansys Fluent uses the PDF approach.

Important:

The PDF method described here applies to the NO_x transport equations only. The preceding combustion simulation can use either the generalized finite-rate chemistry model by Magnussen and Hjertager, the non-premixed or partially premixed combustion model. For details on these models, refer to Species Transport and Finite-Rate Chemistry (p. 237), Non-Premixed Combustion (p. 295), and Partially Premixed Combustion (p. 343).

9.1.9.2. The PDF Approach

The PDF method has proven very useful in the theoretical description of turbulent flow [279] (p. 1069). In the Ansys Fluent NO_x model, a single- or joint-variable PDF in terms of a normalized temperature, species mass fraction, or the combination of both is used to predict the NO_x emission. If the non-premixed or partially premixed combustion model is used to model combustion, then a one- or two-variable PDF in terms of mixture fraction(s) is also available. The mean values of the independent variables needed for the PDF construction are obtained from the solution of the transport equations.

9.1.9.3. The General Expression for the Mean Reaction Rate

The mean turbulent reaction rate $\overline{w}$ can be described in terms of the instantaneous rate w and a single or joint PDF of various variables. In general,

$$\overline{w} = \int \ldots \int w(V_1, V_2, \ldots) P(V_1, V_2, \ldots) dV_1 dV_2 \ldots \tag{9.104}$$

where $V_1, V_2, \ldots$ are temperature and/or the various species concentrations present. P is the probability density function (PDF).

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

9.1.9.4. The Mean Reaction Rate Used in Ansys Fluent

The PDF is used for weighting against the instantaneous rates of production of NO (for example, Equation 9.15 (p. 368)) and subsequent integration over suitable ranges to obtain the mean turbulent reaction rate. Hence we have

$$\overline{S}_{\mathrm{NO}}=\int \rho w_{\mathrm{NO}}(V_1)P_1(V_1)dV \tag{9.105}$$

or, for two variables

$$\overline{S}_{\mathrm{NO}}=\int\int \rho w_{\mathrm{NO}}(V_1,V_2)P(V_1,V_2)dV_1dV_2 \tag{9.106}$$

where $\overline{S}_{NO}$ is the mean turbulent rate of production of NO, w_{NO} is the instantaneous molar rate of NO production, ρ is the instantaneous density, and $P_1(V_1)$ and $P(V_1,V_2)$ are the PDFs of the variables V_1 and, if relevant, V_2. The same treatment applies for the HCN or NH_3 source terms.

Equation 9.105 (p. 392) or Equation 9.106 (p. 392) must be integrated at every node and at every iteration. For a PDF in terms of temperature, the limits of integration are determined from the minimum and maximum values of temperature in the combustion solution (note that you have several options for how the maximum temperature is calculated, as described in Setting Turbulence Parameters in the User's Guide). For a PDF in terms of mixture fraction, the limits of the integrations in Equation 9.105 (p. 392) or Equation 9.106 (p. 392) are determined from the values stored in the look-up tables.

9.1.9.5. Statistical Independence

In the case of the two-variable PDF, it is further assumed that the variables V_1 and V_2 are statistically independent, so that $P(V_1,V_2)$ can be expressed as

$$P(V_1,V_2)=P_1(V_1)P_2(V_2) \tag{9.107}$$

9.1.9.6. The Beta PDF Option

Ansys Fluent can assume P to be a two-moment beta function that is appropriate for combustion calculations [230] (p. 1066), [441] (p. 1078). The equation for the beta function is

$$P(V)=\frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}V^{\alpha-1}(1-V)^{\beta-1}=\frac{V^{\alpha-1}(1-V)^{\beta-1}}{\int_0^1 V^{\alpha-1}(1-V)^{\beta-1}dV} \tag{9.108}$$

where $\Gamma()$ is the Gamma function, and α and β depend on the mean value of the quantity in question, $\overline{m}$, and its variance, σ^2:

$$\alpha=\overline{m}\left(\frac{\overline{m}(1-\overline{m})}{\sigma^2}-1\right) \tag{9.109}$$

$$\beta=(1-\overline{m})\left(\frac{\overline{m}(1-\overline{m})}{\sigma^2}-1\right) \tag{9.110}$$

The beta function requires that the independent variable V assumes values between 0 and 1. Thus, field variables such as temperature must be normalized. See Setting Turbulence Parameters in the

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

User's Guide for information on using the beta PDF when using single-mixture fraction models and two-mixture fraction models.

9.1.9.7. The Gaussian PDF Option

Ansys Fluent can also assume P to exhibit a clipped Gaussian form with delta functions at the tails.

The cumulative density function for a Gaussian PDF (G_{CDF}) may be expressed in terms of the error function as follows:

$$G_{CDF}=\frac{1}{2}\left(1+erf\left(\left(m-\overline{m}\right)\ /\ \sqrt{2\sigma^2}\right)\right) \tag{9.111}$$

where $erf()$ is the error function, m is the quantity of interest, and $\overline{m}$ and σ^2 are the mean and variance values of m, respectively. The error function may be expressed in terms of the incomplete gamma function ($gammp()$):

$$\begin{aligned} \text{for } m<0: erf(m) &= -gammp(0.5,\ m^2) \\ \text{for } m\geq 0: erf(m) &= gammp(0.5,\ m^2) \end{aligned} \tag{9.112}$$

9.1.9.8. The Calculation Method for the Variance

The variance of the field variables (temperature and species mass fraction) σ^2 can be computed by solving the following transport equation during the combustion calculation or pollutant postprocessing stage:

$$\frac{\partial}{\partial t}\left(\rho\sigma^2\right)+\nabla\cdot\left(\rho\vec{v}\sigma^2\right)=\nabla\left(\frac{\mu_t}{\sigma_t}\nabla\sigma^2\right)+C_g\mu_t\left(\nabla\overline{m}\right)^2-C_d\rho\frac{\varepsilon}{k}\sigma^2 \tag{9.113}$$

where the constants σ_t, C_g, and C_d take the values 0.85, 2.86, and 2.0, respectively.

Note that the above variance transport equation is solved only for the temperature variance. Solving an additional equation may be computationally intensive, and therefore may not always be a preferred option for a postprocessing treatment of NO_x prediction.

In addition to the variance transport equation, Ansys Fluent provides an option to calculate σ^2 algebraically. This is an approximate method which assumes equal production and dissipation of variance, and is as follows:

$$\sigma^2=\frac{\mu_t}{\rho}\frac{k}{\varepsilon}\frac{C_g}{C_d}\left(\nabla\overline{m}\right)^2=\frac{\mu_t}{\rho}\frac{k}{\varepsilon}\frac{C_g}{C_d}\left[\left(\frac{\partial\overline{m}}{\partial x}\right)^2+\left(\frac{\partial\overline{m}}{\partial y}\right)^2+\left(\frac{\partial\overline{m}}{\partial z}\right)^2\right] \tag{9.114}$$

The term in brackets is the dissipation rate of the independent variable. The algebraic option is a preferred choice when PDF is constructed with more than one variable (like temperature-species PDF in Equation 9.107 (p. 392)). When the turbulence chemistry interaction mode is temperature-species, the species variance is computed algebraically, while the temperature variance can be computed either algebraically or by using the transport equation.

For a PDF in terms of mixture fraction, the mixture fraction variance has already been solved as part of the basic combustion calculation, so no additional calculation for σ^2 is required.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

9.2. Soot Formation

Information about the theory behind soot formation is presented in the following sections. For information about using soot formation models in Ansys Fluent, see Using the Soot Models in the User's Guide.

9.2.1. Overview and Limitations

9.2.2. Soot Model Theory

9.2.1. Overview and Limitations

Ansys Fluent provides five models for the prediction of soot formation in combustion systems. The predicted soot concentration can be coupled with radiation absorption for the P-1, discrete ordinates, or discrete transfer radiation models with a variable absorption coefficient.

9.2.1.1. Predicting Soot Formation

Ansys Fluent predicts soot concentrations in a combustion system using one of the following available models:

- the one-step Khan and Greeves model [300] (p. 1070), in which Ansys Fluent predicts the rate of soot formation based on an empirical relation. This is a default soot formation model in Ansys Fluent.
- the two-step Tesner model [407] (p. 1076), [649] (p. 1090), in which Ansys Fluent predicts the formation of nuclei particles and computes soot formation on the nuclei. In this and previous models, combustion of the soot (and particle nuclei) is assumed to be governed by the Magnussen combustion rate [407] (p. 1076). This assumption limits the use of these models to turbulent flows. The detailed chemistry and physics of soot formation are quite complex and both of these models are empirically-based and can be used to approximate soot formation process in combustion systems. The results yielded by these models should be considered as qualitative indicators of the system performance, unless validated by experimental results.
- the Moss-Brookes model [81] (p. 1057), in which Ansys Fluent predicts soot formation for methane flames (and higher hydrocarbon species, if appropriately modified) by solving transport equations for normalized radical nuclei concentration and the soot mass fraction. This model has less empiricism and should theoretically provide better accuracy than the Khan and Greeves and Tesner models.
- the Moss-Brookes-Hall model [227] (p. 1066), which is an extension of the Moss-Brookes model and is applicable for higher hydrocarbon fuels (for example, kerosene). The Hall extension provides further options for modeling higher hydrocarbon fuels. Note that the Moss-Brookes-Hall model is only available when the required species are present in the gas phase species list.
- the Method of Moments model [192] (p. 1064), in which Ansys Fluent predicts soot formation based on the soot particle population balance methodology. The Method of Moments considers a soot size distribution where the diameters of the soot particles are dynamically evolving. This approach uses fewer empirical constants for modeling of various soot formation sub-processes, such as nucleation, coagulation, and the surface growth kernels.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

9.2.1.2. Restrictions on Soot Modeling

The following restrictions apply to soot formation models:

- You must use the pressure-based solver. The soot models are not available with either of the density-based solvers.
- The Khan and Greeves model and the Tesner model can model soot formation only for turbulent flows whereas the Moss-Brookes model, the Moss-Brookes-Hall model, and the Method of Moments model can be used with both laminar and turbulent flows.
- The soot model cannot be used in conjunction with the premixed combustion model.

9.2.2. Soot Model Theory

9.2.2.1. The One-Step Soot Formation Model

In the one-step Khan and Greeves model [300] (p. 1070), Ansys Fluent solves a single transport equation for the soot mass fraction:

$$\frac{\partial}{\partial t}\left(\rho Y_{soot}\right)+\nabla\cdot\left(\rho\vec{v}Y_{soot}\right)=\nabla\cdot\left(\frac{\mu_t}{\sigma_{soot}}\nabla Y_{soot}\right)+\mathcal{R}_{soot} \tag{9.115}$$

where

Y_{soot} = soot mass fraction

σ_{soot} = turbulent Prandtl number for soot transport

$\mathcal{R}_{soot}$ = net rate of soot generation (kg/m^3–s)

$\mathcal{R}_{soot}$, the net rate of soot generation, is the balance of soot formation, $\mathcal{R}_{soot,form}$, and soot combustion, $\mathcal{R}_{soot,comb}$:

$$\mathcal{R}_{soot}=\mathcal{R}_{soot,form}-\mathcal{R}_{soot,comb} \tag{9.116}$$

The rate of soot formation is given by a simple empirical rate expression:

$$\mathcal{R}_{soot,form}=C_s p_{fuel}\phi^r e^{-E/RT} \tag{9.117}$$

where

C_s = soot formation constant (kg/N-m-s)

p_{fuel} = fuel partial pressure (Pa)

ϕ = equivalence ratio

r = equivalence ratio exponent

E/R = activation temperature (K)

The rate of soot combustion is the minimum of two rate expressions [407] (p. 1076):

$$\mathcal{R}_{soot,comb}=\min\left[\mathcal{R}_1,\mathcal{R}_2\right] \tag{9.118}$$

The two rates are computed as

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$\mathcal{R}_1 = A\rho Y_{soot}\frac{\varepsilon}{k} \tag{9.119}$$

and

$$\mathcal{R}_2 = A\rho\left(\frac{Y_{ox}}{\nu_{soot}}\right)\left(\frac{Y_{soot}\nu_{soot}}{Y_{soot}\nu_{soot}+Y_{fuel}\nu_{fuel}}\right)\frac{\varepsilon}{k} \tag{9.120}$$

where

A = constant in the Magnussen model

Y_{ox}, Y_{fuel} = mass fractions of oxidizer and fuel

ν_{soot}, ν_{fuel} = mass stoichiometries for soot and fuel combustion

The default constants for the one-step model are valid for a wide range of hydrocarbon fuels.

9.2.2.2. The Two-Step Soot Formation Model

The two-step Tesner model [649] (p. 1090) predicts the generation of radical nuclei and then computes the formation of soot on these nuclei. Ansys Fluent therefore solves transport equations for two scalar quantities: the soot mass fraction (Equation 9.115 (p. 395)) and the normalized radical nuclei concentration:

$$\frac{\partial}{\partial t}\left(\rho b^*_{nuc}\right)+\nabla\cdot\left(\rho\vec{v}b^*_{nuc}\right)=\nabla\cdot\left(\frac{\mu_t}{\sigma_{nuc}}\nabla b^*_{nuc}\right)+\mathcal{R}^*_{nuc} \tag{9.121}$$

where

b^*_{nuc} = normalized radical nuclei concentration (number of particles $\times 10^{-15}$/kg)

σ_{nuc} = turbulent Prandtl number for nuclei transport

$\mathcal{R}^*_{nuc}$ = normalized net rate of nuclei generation (number of particles $\times 10^{-15}$/m^3–s)

In these transport equations, the rates of nuclei and soot generation are the net rates, involving a balance between formation and combustion.

9.2.2.2.1. Soot Generation Rate

The two-step model computes the net rate of soot generation, $\mathcal{R}_{soot}$, in the same way as the one-step model, as a balance of soot formation and soot combustion:

$$\mathcal{R}_{soot}=\mathcal{R}_{soot,form}-\mathcal{R}_{soot,comb} \tag{9.122}$$

In the two-step model, however, the rate of soot formation, $\mathcal{R}_{soot,form}$, depends on the concentration of radical nuclei, c_{nuc}:

$$\mathcal{R}_{soot,form}=m_p\left(\alpha-\beta N_{soot}\right)c_{nuc} \tag{9.123}$$

where

m_p = mean mass of soot particle (kg/number of particles)

N_{soot} = concentration of soot particles (number of particles/m^3)

c_{nuc} = radical nuclei concentration = ρb_{nuc} (number of particles/m^3)

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

α = empirical constant (s^{-1})

β = empirical constant (m^3/number of particles-s)

The rate of soot combustion, $\mathcal{R}_{soot,comb}$, is computed in the same way as for the one-step model, using Equation 9.118 (p. 395) – Equation 9.120 (p. 396).

The default constants for the two-step model are applicable for the combustion of acetylene (C_2H_2). According to Ahmad et al. [10] (p. 1053), these values should be modified for other fuels, as the sooting characteristics of acetylene are known to be different from those of saturated hydrocarbon fuels.

9.2.2.2.2. Nuclei Generation Rate

The net rate of nuclei generation in the two-step model is given by the balance of the nuclei formation rate and the nuclei combustion rate:

$$\mathcal{R}^*_{nuc} = \mathcal{R}^*_{nuc,form} - \mathcal{R}^*_{nuc,comb} \tag{9.124}$$

where

$\mathcal{R}^*_{nuc,form}$ = rate of nuclei formation (number of particles $\times 10^{-15}$/m^3-s)

$\mathcal{R}^*_{nuc,comb}$ = rate of nuclei combustion (number of particles $\times 10^{-15}$/m^3-s)

The rate of nuclei formation, $\mathcal{R}^*_{nuc,form}$, depends on a spontaneous formation and branching process, described by

$$\mathcal{R}^*_{nuc,form} = \eta_0 + (f-g)c^*_{nuc} - g_0 c^*_{nuc} N_{soot} \tag{9.125}$$

$$\eta_0 = a^*_0 c_{fuel} e^{-E/RT} \tag{9.126}$$

where

c^*_{nuc} = normalized nuclei concentration ($=\rho b^*_{nuc}$)

$a^*_0 = a_0 / 10^{15}$

a_0 = pre-exponential rate constant (number of particles/kg-s)

c_{fuel} = fuel concentration (kg/m^3)

$f-g$ = linear branching − termination coefficient (s^{-1})

g_0 = linear termination on soot particles (m^3/number of particles-s)

Note that the branching term in Equation 9.125 (p. 397), $(f-g)c^*_{nuc}$, is included only when the kinetic rate, η_0, is greater than the limiting formation rate (10^5 number of particles/m^3-s, by default).

The rate of nuclei combustion is assumed to be proportional to the rate of soot combustion:

$$\mathcal{R}^*_{nuc,comb} = \mathcal{R}_{soot,comb} \frac{b^*_{nuc}}{Y_{soot}} \tag{9.127}$$

where the soot combustion rate, $\mathcal{R}_{soot,comb}$, is given by Equation 9.118 (p. 395).

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

9.2.2.3. The Moss-Brookes Model

The Moss-Brookes model solves transport equations of normalized radical nuclei concentration b^*_{nuc} and soot mass fraction Y_{soot}:

$$\frac{\partial}{\partial t}\left(\rho Y_{soot}\right)+\nabla\cdot\left(\rho\vec{v}Y_{soot}\right) = \nabla\cdot\left(\frac{\mu_t}{\sigma_{soot}}\nabla Y_{soot}\right)+\frac{dM}{dt} \tag{9.128}$$

$$\frac{\partial}{\partial t}\left(\rho b^*_{nuc}\right)+\nabla\cdot\left(\rho\vec{v}b^*_{nuc}\right) = \nabla\cdot\left(\frac{\mu_t}{\sigma_{nuc}}\nabla b^*_{nuc}\right)+\frac{1}{N_{norm}}\frac{dN}{dt} \tag{9.129}$$

where

Y_{soot} = soot mass fraction

M = soot mass concentration (kg/m^3)

b^*_{nuc} = normalized radical nuclei concentration (particles $\times 10^{-15}$/kg) = $\frac{N}{\rho N_{norm}}$

N = soot particle number density (particles/m^3)

$N_{norm} = 10^{15}$ particles

The instantaneous production rate of soot particles, subject to nucleation from the gas phase and coagulation in the free molecular regime, is given by

$$\frac{dN}{dt}=\underbrace{C_\alpha N_A\left(\frac{X_{prec}P}{RT}\right)^l\exp\left\{-\frac{T_\alpha}{T}\right\}}_{\text{Nucleation}}-\underbrace{C_\beta\left(\frac{24RT}{\rho_{soot}N_A}\right)^{1/2}d_p^{1/2}N^2}_{\text{Coagulation}} \tag{9.130}$$

where C_α, C_β and l are model constants. Here, N_A (= 6.022045 x 10^{26}kmol^{-1}) is the Avogadro number and X_{prec} is the mole fraction of soot precursor (for methane, the precursor is assumed to be acetylene, whereas for kerosene it is a combination of acetylene and benzene). The mass density of soot, ρ_{soot}, is assumed to be 1800 kg/m^3 and d_p is the mean diameter of a soot particle. The nucleation rate for soot particles is taken to be proportional to the local acetylene concentration for methane. The activation temperature T_α for the nucleation reaction is proposed by Lindstedt [375] (p. 1074).

The source term for soot mass concentration is modeled by the expression

$$\begin{aligned}\frac{dM}{dt} =& \underbrace{M_P C_\alpha\left(\frac{X_{prec}P}{RT}\right)^l\exp\left\{-\frac{T_\alpha}{T}\right\}}_{\text{Nucleation}}\\ &+\underbrace{C_\gamma\left(\frac{X_{sgs}P}{RT}\right)^m\exp\left\{-\frac{T_\gamma}{T}\right\}\left[(\pi N)^{1/3}\left(\frac{6M}{\rho_{soot}}\right)^{2/3}\right]^n}_{\text{Surface Growth}}\\ &-\underbrace{C_{oxid}C_\omega\eta_{coll}\left(\frac{X_{OH}P}{RT}\right)\sqrt{T}\,(\pi N)^{1/3}\left(\frac{6M}{\rho_{soot}}\right)^{2/3}}_{\text{Oxidation}}\end{aligned} \tag{9.131}$$

where C_γ, C_{oxid}, C_ω, m, and n are additional model constants. The constant M_P (= 144 kg/kmol) is the mass of an incipient soot particle, here taken to consist of 12 carbon atoms. Even though the model is not found to be sensitive to this assumption, a nonzero initial mass is needed to begin the process of surface growth. Here, X_{sgs} is the mole fraction of the participating surface growth species. For paraffinic fuels, soot particles have been found to grow primarily by the addition of

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

gaseous species at their surfaces, particularly acetylene that has been found in abundance in the sooting regions of laminar methane diffusion flames.

The model assumes that the hydroxyl radical is the dominant oxidizing agent in methane/air diffusion flames and that the surface-specific oxidation rate of soot by the OH radical may be formulated according to the model proposed by Fenimore and Jones [173] (p. 1062). Assuming a collision efficiency (η_{coll}) of 0.04, the oxidation rate may be written as Equation 9.131 (p. 398).

The process of determination of the exponents l, m, and n are explained in detail by Brookes and Moss [81] (p. 1057). The constants C_α and C_β are determined through numerical modeling of a laminar flame for which experimental data exists.

The set of constants proposed by Brookes and Moss for methane flames are given below:

C_α = 54 s^{-1} (model constant for soot inception rate)

T_α = 21000 K (activation temperature of soot inception)

C_β = 1.0 (model constant for coagulation rate)

C_γ = 11700 kg m $kmol^{-1}$ s^{-1} (surface growth rate scaling factor)

T_γ = 12100 K (activation temperature of surface growth rate)

C_ω = 105.8125 kg m $kmol^{-1}$ $K^{1/2}$ s^{-1} (oxidation model constant)

η_{coll} = 0.04 (collisional efficiency parameter)

C_{oxid} = 0.015 (oxidation rate scaling parameter)

Note that the implementation of the Moss-Brookes model in Ansys Fluent uses the values listed above, except for C_{oxid}, which is set to unity by default.

The closure for the mean soot source terms in the above equations was also described in detail by Brookes and Moss [81] (p. 1057). The uncorrelated closure is the preferred option for a tractable solution of the above transport equations.

Moss et al. [460] (p. 1079) have shown the above model applied to kerosene flames by modifying only the soot precursor species (in the original model the precursor was acetylene, whereas for kerosene flames the precursor was assumed to be a combination of both acetylene and benzene) and by setting the value of oxidation scaling parameter C_{oxid} to unity. A good comparison against the experimental measurements for the lower pressure (7 bar) conditions was observed. The predictions of soot formation within methane flames have shown the Brooks and Moss [81] (p. 1057) model to be superior compared with the standard Tesner et al. [649] (p. 1090) formulation.

9.2.2.3.1. The Moss-Brookes-Hall Model

Since the Moss-Brookes model was mainly developed and validated for methane flames, a further extension for higher hydrocarbon fuels called the Moss-Brookes-Hall model is available in Ansys Fluent. Here, the extended version is a model reported by Wen et al. [698] (p. 1093), based on model extensions proposed by Hall et al. [227] (p. 1066) and an oxidation model proposed by Lee et al. [347] (p. 1073). The work of Hall [227] (p. 1066) is based on a soot inception rate due to two-ringed and three-ringed aromatics, as opposed to the Moss-Brookes assumption of a soot inception due to acetylene or benzene (for higher hydrocarbons).

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Hall et al. [227] (p. 1066) proposed a soot inception rate based on the formation rates of two-ringed and three-ringed aromatics ($C_{10}H_7$ and $C_{14}H_{10}$), from acetylene (C_2H_2), benzene (C_6H_6), and the phenyl radical (C_6H_5) based on the following mechanisms:

$$\begin{aligned} 2\,C_2H_2 + C_6H_5 &\rightleftharpoons C_{10}H_7 + H_2 \\ C_2H_2 + C_6H_6 + C_6H_5 &\rightleftharpoons C_{14}H_{10} + H + H_2 \end{aligned} \tag{9.132}$$

Based on their laminar methane flame data, the inception rate of soot particles was given to be eight times the formation rate of species $C_{10}H_7$ and $C_{14}H_{10}$, as shown by

$$\begin{aligned} \left(\frac{dN}{dt}\right)_{inception} = \; & 8C_{\alpha,1}\frac{N_A}{M_P}\left[\rho_{soot}^2\left(\frac{Y_{C_2H_2}}{W_{C_2H_2}}\right)^2 Y_{C_6H_5} W_{H_2}\right]\exp\left\{-\frac{T_{\alpha,1}}{T}\right\} \\ & + 8C_{\alpha,2}\frac{N_A}{M_P}\left[\rho_{soot}^2\left(\frac{Y_{C_2H_2}}{W_{C_2H_2}}\right)^2 Y_{C_6H_5} W_{H_2}\right]\exp\left\{-\frac{T_{\alpha,2}}{T}\right\} \end{aligned} \tag{9.133}$$

where $C_{\alpha,1}$ = 127 x $10^{8.88}$ s^{-1}, $C_{\alpha,2}$ = 178 x $10^{9.50}$ s^{-1}, $T_{\alpha,1}$ = 4378 K, and $T_{\alpha,2}$ = 6390 K as determined by Hall et al. [227] (p. 1066). In their model, the mass of an incipient soot particle was assumed to be 1200 kg/kmol (corresponding to 100 carbon atoms, as opposed to 12 carbon atoms used by Brookes and Moss [81] (p. 1057)). The mass density of soot was assumed to be 2000 kg/m^3, which is also slightly different from the value used by Brookes and Moss [81] (p. 1057).

Both the coagulation term and the surface growth term were formulated similar to those used by Brookes and Moss [81] (p. 1057), with a slight modification to the constant C_γ so that the value is 9000.6 kg m $kmol^1$ s^{-1} (based on the model developed by Lindstedt [376] (p. 1074)).

For the soot oxidation term, oxidation due to O_2 (based on measurements and a model based on Lee et al. [347] (p. 1073)) was added, in addition to the soot oxidation due to the hydroxyl radical. By assuming that the kinetics of surface reactions are the limiting mechanism and that the particles are small enough to neglect the diffusion effect on the soot oxidation, they derived the specific rate of soot oxidation by molecular oxygen. Therefore, the full soot oxidation term, including that due to hydroxyl radical, is of the form

$$\begin{aligned} \left(\frac{dM}{dt}\right)_{oxid} = \; & -C_{oxid}C_{\omega,1}\eta_{coll}\left(\frac{X_{OH}P}{RT}\right)\sqrt{T}\,(\pi N)^{1/3}\left(\frac{6M}{\rho_{soot}}\right)^{2/3} \\ & -C_{oxid}C_{\omega,2}\left(\frac{X_{O_2}P}{RT}\right)\exp\left\{-\frac{T_{\omega,2}}{T}\right\}\sqrt{T}\,(\pi N)^{1/3}\left(\frac{6M}{\rho_{soot}}\right)^{2/3} \end{aligned} \tag{9.134}$$

Here, the collision efficiency is assumed to be 0.13 (compared to the value of 0.04 used by Brookes and Moss) and the oxidation rate scaling parameter is set to 0.015. The model constants used are as follows:

$C_{\omega,1}$ = 105.81 kg m $kmol^1$ $K^{1/2}$ s^{-1} (same as that used by Brookes and Moss)

$C_{\omega,2}$ =8903.51 kg m $kmol^1$ $K^{1/2}$ s^{-1}

$T_{\omega,2}$ =19778 K

9.2.2.3.2. Soot Formation in Turbulent Flows

The kinetic mechanisms of soot formation and destruction for the Moss-Brookes model and the Hall extension are obtained from laboratory experiments, in a similar fashion to the NO_x model. In any practical combustion system, however, the flow is highly turbulent. The turbulent mixing

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

process results in temporal fluctuations in temperature and species concentration that will influence the characteristics of the flame.

The relationships among soot formation rate, temperature, and species concentration are highly nonlinear. Hence, if time-averaged composition and temperature are employed in any model to predict the mean soot formation rate, significant errors will result. Temperature and composition fluctuations must be taken into account by considering the probability density functions that describe the time variation.

9.2.2.3.2.1. The Turbulence-Chemistry Interaction Model

In turbulent combustion calculations, Ansys Fluent solves the density-weighted time-averaged Navier-Stokes equations for temperature, velocity, and species concentrations or mean mixture fraction and variance. To calculate soot concentration for the Moss-Brookes model and the Hall extension, a time-averaged soot formation rate must be computed at each point in the domain using the averaged flow-field information.

9.2.2.3.2.2. The PDF Approach

The PDF method has proven very useful in the theoretical description of turbulent flow [279] (p. 1069). In the Ansys Fluent Moss-Brookes model and the Hall extension, a single- or joint-variable PDF in terms of a normalized temperature, species mass fraction, or the combination of both is used to predict the soot formation. If the non-premixed combustion model is used to model combustion, then a one- or two-variable PDF in terms of mixture fraction(s) is also available. The mean values of the independent variables needed for the PDF construction are obtained from the solution of the transport equations.

9.2.2.3.2.3. The Mean Reaction Rate

The mean turbulent reaction rate described in The General Expression for the Mean Reaction Rate (p. 391) for the NO_x model also applies to the Moss-Brookes model and the Hall extension. The PDF is used for weighting against the instantaneous rates of production of soot and subsequent integration over suitable ranges to obtain the mean turbulent reaction rate as described in Equation 9.105 (p. 392) and Equation 9.106 (p. 392) for NO_x.

9.2.2.3.2.4. The PDF Options

As is the case with the NO_x model, P can be calculated as either a two-moment beta function or as a clipped Gaussian function, as appropriate for combustion calculations [230] (p. 1066), [441] (p. 1078). Equation 9.108 (p. 392) – Equation 9.112 (p. 393) apply to the Moss-Brookes model and Hall extension as well, with the variance σ^2 computed by solving a transport equation during the combustion calculation stage, using Equation 9.113 (p. 393) or Equation 9.114 (p. 393).

9.2.2.3.3. The Effect of Soot on the Radiation Absorption Coefficient

A description of the modeling of soot-radiation interaction is provided in The Effect of Soot on the Absorption Coefficient (p. 220).

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

9.2.2.4. The Method of Moments Model

To accurately predict soot formation, the detailed kinetic modeling of soot formation and resolution of the particle-size distribution are required. The method of moments, where only few moments are solved, is a computationally efficient approach for modeling soot formation.

9.2.2.4.1. Soot Particle Population Balance

During soot formation, the soot particles are present in a wide range of diameters. The particle population balance method uses the particle-size distribution (PSD) for modeling of the soot particle surface area involved in the calculations of important sub steps of soot formation process like soot particles surface growth and oxidation. The various stages of soot formation have an impact on the soot PSD. The evolution of the soot particles' population can be represented by the population balance equations, also known as the Smoluchowski master equations:

$$\frac{dN_1}{dt} = -\sum_{j=1}^{\infty} \beta_{1,j} N_1 N_j \tag{9.135}$$

$$\frac{dN_i}{dt} = \frac{1}{2}\sum_{j=1}^{i-1} \beta_{j,i-j} N_{i-j} N_j - \sum_{j=1}^{\infty} \beta_{i,j} N_i N_j$$

where N_i is the particle number density function that denotes the number of the soot particles of the i^{th} size class per unit volume, $\beta_{i,j}$ is the collision coefficient between particles of size classes i and j.

For the first particle size class (the first equation), the right-hand side contains only one negative source term. This term reflects the reduction in the first size class population due to coagulation between the particles of the current size class and any other particles resulting in formation of particles of higher size classes. For all other higher particle size classes, the positive source term appears in the right-hand side of the second equation indicating the growth of the particle number density function N_i as a result of said coagulation between particles of lower size classes.

The principal difficulty in obtaining a solution for Equation 9.135 (p. 402) involves solving an infinite number of particle size classes. The two approaches commonly used for these problems are the sectional method and the method of moments.

The sectional method is based on discretization of the entire range of particle size classes into a predefined finite number of intervals. Considering that the particle mass and population will be spread over all intervals and will be also affected by other processes (such as, surface reactions and oxidation), a sufficiently large number of sections (or finite intervals) is required for obtaining an accurate solution. For these reasons, applying the sectional method to Equation 9.135 (p. 402) is computationally demanding. An efficient alternative to the sectional method is the method of moments. This method considers moments of the soot PSD functions.

The discrete form of the r^{th} moment of particle-size distribution function N_i for the particle of the size class i is defined as:

$$M_r = \sum_{i=1}^{\infty} i^r N_i \tag{9.136}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

From Equation 9.136 (p. 402), the total particle number of the particle population is equal to zeroth moment:

$$N_t = \sum_{i=1}^{\infty} N_i = M_0 = Total\ number\ of\ particles$$

The mass of a particle is proportional to its class; therefore, the total mass of the particle population is calculated as:

$$m_{p,t} = \sum_{i=1}^{\infty} m_0 i N_i = m_0 \sum_{i=1}^{\infty} i^1 N_i = m_0 M_1 = Total\ mass\ of\ particles$$

where m_0 is a constant that represents the mass of a bulk species molecule comprising the particle core.

Similarly, the total particles volume can be obtained by:

$$v_{p,t} = \frac{m_0 M_1}{\rho_b}$$

where ρ_b is the bulk density of particle core (soot density).

Multiplying Equation 9.135 (p. 402) by the soot particle size class i and using the expression of Equation 9.136 (p. 402), we obtain:

$$\frac{d}{dt}\left[\sum_{i=1}^{\infty} i^r N_i\right] = \frac{dM_r}{dt} = G_r, \quad \text{for } r = 0, 1, 2... \tag{9.137}$$

where G_r is the right-hand side of Equation 9.135 (p. 402), which is a coagulation source term.

In soot formation, in addition to the coagulation process, the nucleation and the surface reactions are the two other important contributing factors. In the presence of nucleation and surface reactions, Equation 9.137 (p. 403) is written as

$$\frac{dM_r}{dt} = R_r + G_r + W_r \tag{9.138}$$

where R_r is the nucleation source term, and W_r is the source term due to surface reactions including surface growth and oxidation.

9.2.2.4.2. Moment Transport Equations

Equation 9.138 (p. 403) describes the change in the moments due to various sub-processes at a single point. In a flow system, the moments will also be affected by convection and diffusion. For flow systems, the transport equation for the moments of soot concentration can be written as:

$$\frac{\partial(\rho M_r)}{\partial t} + \nabla.(\rho \vec{v} M_r) = \nabla.\left(\frac{\mu_{eff}}{\sigma_t} \nabla M_r\right) + S_r \tag{9.139}$$

where

M_r = r-th moment of soot size distribution

μ_{eff} = effective diffusion coefficient

σ_t = turbulent Prandtl number for moment transport equation

S_r = source term in the moment transport computed using Equation 9.138 (p. 403)

The source term for computation of S_r is described in the sections that follow.

9.2.2.4.3. Nucleation

In Ansys Fluent, the nucleation process is modeled as coagulation between two soot precursor species. The soot precursor is a user-defined gas phase species. Typically, the soot precursors are poly-cyclic aromatic hydrocarbons (PAH), and the soot nuclei formation is modeled as coagulation of two PAH molecules. The mean diameter of the PAH is calculated from the number of carbon atoms in it and the soot density. The concentration of the PAH is obtained from the gas phase mechanism. In many practical cases, where the chemical mechanism used in simulations is small and does not include PAH species, the smaller species, such as C_2H_2, which is a building block of PAH molecules, can be used as a precursor species.

The nucleation source term for the first moment is calculated as:

$$R_0 = C_{nuc}\sqrt{T}\left[Xp\right]^2 \tag{9.140}$$

where $\left[Xp\right]$ is the molar concentration of the precursor species, and C_{nuc} is the constant calculated by:

$$C_{nuc} = \varepsilon\sqrt{\frac{4\pi k_B}{m_c N_{C,P}}}\left(d_P N_A\right)^2 \tag{9.141}$$

where

ε = 2.2 = van der Waals enhancement factor

k_B = Boltzmann constant

d_P = diameter of precursor species

$N_{C,P}$ = number of carbon atoms in a precursor molecule

m_c = mass of a carbon atom (12 amu)

N_A = Avogadro number

The nucleation rates calculated based on the kinetic theory using Equation 9.140 (p. 404) are generally very large. To properly scale them, Equation 9.140 (p. 404) is adjusted using a sticking coefficient γ:

$$R_0 = \gamma C_{nuc}\sqrt{T}\left[Xp\right]^2 \tag{9.142}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

The value of the sticking coefficient varies with the size of the precursor species (molecular weight). Table 9.4: Sticking Coefficient for Different PAH Species (p. 405) lists the sticking coefficient values for different precursors as proposed by Blanquart and Pitsch [64] (p. 1056).

Table 9.4: Sticking Coefficient for Different PAH Species

Species Name	Formula	Molecular Weight (gr/mol)	γ_i
naphthalene	$C_{10}H_8$	128	0.0010
acenaphthylene	$C_{12}H_8$	152	0.0030
biphenyl	$C_{12}H_{10}$	154	0.0085
phenathrene	$C_{14}H_{10}$	178	0.0150
acephenanthrylene	$C_{16}H_{10}$	202	0.0250
pyrene	$C_{16}H_{10}$	202	0.0250
fluoranthene	$C_{16}H_{10}$	202	0.0250
cyclo[cd]pyrene	$C_{18}H_{10}$	226	0.0390

Other suggestions found in the literature propose using the sticking coefficient that is proportional to the fourth power of the precursor molecular weight:

$$\gamma_i = C_N m_i^4 \tag{9.143}$$

where C_N is the constant that can be either calculated using curve fitting from Table 9.4: Sticking Coefficient for Different PAH Species (p. 405) or obtained from experimental data. Equation 9.143 (p. 405) could be used to approximate the sticking coefficient values for precursors that are not listed in Table 9.4: Sticking Coefficient for Different PAH Species (p. 405).

Alternatively, the nucleation can also be specified as an irreversible kinetic reaction between two precursors:

$$\nu_1 PAH_1 + \nu_2 PAH_2 \rightarrow Soot + \mathrm{Products} \tag{9.144}$$

$$R_0 = AT^B \exp\left(-E/RT\right)\left[PAH_1\right]^{\eta_1}\left[PAH_2\right]^{\eta_2} \tag{9.145}$$

where η and ν are the power exponent and stoichiometric coefficients of the precursor species.

Equation 9.145 (p. 405) is analogues to Equation 9.140 (p. 404) having the same PAH species and with the following values for the constants:

$$A = \gamma C_{num}$$
$$B = 0.5$$
$$E = 0$$
$$\eta_1 = \eta_2 = 0.5$$
$$\nu_1 = \nu_2 = 1$$

The nucleation source terms for higher moments are calculated from the lower moment source terms using the following expression:

$$R_r = 2N_{C,P}R_{r-1}, \quad \text{for } r > 0 \tag{9.146}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

9.2.2.4.4. Coagulation

Once formed, the soot particles collide with each other affecting the size distribution of the soot particles population. The process of coagulation assumes that the resulting particle remains a sphere with an increased diameter. The coagulation process changes the number density but not the total mass of the particle. The source terms in the moment transport equation due to coagulation are calculated as:

$$G_0 = -0.5 \sum_{i=1}^{\infty} \sum_{j=1}^{\infty} \beta_{i,j} N_i N_j \tag{9.147}$$

The coagulation term for the second size class G_1 is equal to 0 because the coagulation does not change the total mass of soot formed.

The coagulation terms for higher size classes are calculated by:

$$G_r = 0.5 \sum_{k=1}^{r-1} \binom{r}{k} \sum_{i=1}^{\infty} \sum_{j=1}^{\infty} i^k j^{r-k} \beta_{i,j} N_i N_j \quad for\ r \geq 2 \tag{9.148}$$

where $\beta_{i,j}$ is the collision efficiency, which is dependent on the coagulation regimes.

The coagulation process can take place in different regimes:

- continuum
- free molecular
- intermediate of the two (or transition)

The regimes of coagulation depends on the Knudsen number

$$Kn = \lambda / d_s \tag{9.149}$$

where λ the gas mixture mean free path, and d_s is the soot particle diameter. The mean free path of soot particles λ are calculated as follows:

$$\lambda = \frac{k_B T}{\sqrt{2}\, \pi d_{gas}^2 P} \tag{9.150}$$

where k_B is the Boltzmann constant, P is the pressure, and d_{gas} is the diameter of the gas molecules calculated by

$$d_{gas} = \left(\frac{6M}{\rho \pi N_A} \right)^{\frac{1}{3}} \tag{9.151}$$

where M is the molecular weight of the gas.

The mean diameter of the soot particles d_s is calculated using the definition of the soot moments in Equation 9.136 (p. 402) in the following manner:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$d_s = \frac{d_1 \sum_{i=1}^{\infty} i^{1/3} N_i}{\sum_{i=1}^{\infty} N_i} = d_1 \frac{M_{1/3}}{M_0} \tag{9.152}$$

where d_1 is the diameter of smallest particle, which is a single carbon atom.

For different regimes, different treatments are applied when solving Equation 9.148 (p. 406) as further described.

Continuum Coagulation ($K_n < 0.1$)

The collision coefficient in the continuum region is expressed as:

$$\beta_{i,j}^{c} = K_C \left(\frac{C_i}{i^{1/3}} + \frac{C_j}{j^{1/3}} \right) \left(i^{1/3} + j^{1/3} \right) \tag{9.153}$$

where C_i is the Cunningham slip correction factor equal to 1+1.257 Kn, K_C is the continuum collision factor calculated as:

$$K_C = \frac{2k_B T}{3\eta}$$

where η is the molecular viscosity of the mixture.

Substituting the collision coefficient (defined in Equation 9.153 (p. 407)) into Equation 9.148 (p. 406) gives the source terms due to coagulation described below.

For the first moment, the moment source term due to continuum coagulation is:

$$G_0^c = -K_c \left[M_0^2 + M_{1/3} M_{-1/3} + K_c' \left(M_{1/3} M_{-1/3} + M_{1/3} M_{2/3} \right] \right. \tag{9.154}$$

where $K_c' = 1.257\lambda / d_s$.

By introducing the concept of reduced moment $\mu_r = M_r / M_0$, Equation 9.154 (p. 407) can be rewritten as:

$$G_0^c = -K_c \left[1 + \mu_{1/3} \mu_{-1/3} + K_c' \left(\mu_{1/3} \mu_{-1/3} + \mu_{1/3} \mu_{2/3} \right] \right. M_0^2 \tag{9.155}$$

Similarly, for $r = 2, 3 \ldots$. the moment source terms due to continuum coagulation are expressed as:

$$G_r^c = 0.5 K_c \sum_{k=1}^{r-1} \binom{r}{k} \left[\begin{matrix} 2\mu_k \mu_{r-k} + \mu_{k+1/3} \mu_{r-k-1/3} + \mu_{k-1/3} \mu_{r-k+1/3} + K_c' \\ \left(\mu_{k-1/3} \mu_{r-k} + \mu_k \mu_{r-k-1/3} + \mu_{k+1/3} \mu_{r-k-2/3} + \mu_{k-2/3} \mu_{r-k+1/3} \right) \end{matrix} \right] M_0^2 \tag{9.156}$$

The coagulation terms in Equation 9.155 (p. 407) and Equation 9.156 (p. 407) consist of the fractional order moments as well as the negative order moments that need to be calculated for closure. Here, the process of interpolative closure proposed by Frenklach ([192] (p. 1064)) has been used. In the interpolative closure, the positive fractional order moments are obtained using a polynomial interpolation. Since the numerical values of the subsequent moments differ by an order of

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

magnitude, the logarithmic interpolation is used to minimize interpolation errors. The fractional moments are obtained using the following expression:

$$\log\left(\mu_r\right)=L_{Int}\left(0,\log\mu_1,\log\mu_2,\log\mu_3,...\right) \tag{9.157}$$

where, L_{Int} is the Lagrange interpolating polynomial.

In order to calculate the negative order fractional moments, an extrapolation from integer order moments is used

$$\log\left(\mu_r\right)=L_{Ext}\left(0,\log\mu_1,\log\mu_2,\log\mu_3,...\right) \tag{9.158}$$

where L_{Ext} is the Lagrange extrapolation. It has been observed that a quadratic interpolation of Equation 9.157 (p. 408) and a linear extrapolation of equation Equation 9.158 (p. 408) work reasonably well.

Coagulation in the Free Molecular Regime (K_n > 10)

In the free molecular regime, the collision coefficient is calculated as:

$$\beta_{i,j}^{f}=K_f\sqrt{\frac{i+j}{ij}}\left(i^{1/3}+j^{1/3}\right)^2 \tag{9.159}$$

where K_f is the coefficient for the free molecular collision efficiency calculation in the following form:

$$K_f=\varepsilon\sqrt{\frac{6k_BT}{\rho_s}}\left(\frac{3m_c}{4\pi\rho_s}\right)^{1/6} \tag{9.160}$$

Using Equation 9.159 (p. 408) for the collision coefficient, Equation 9.155 (p. 407) and Equation 9.156 (p. 407) give the moment source terms due to coagulation in the free molecular regime.

Note that the collision coefficient written in the form of Equation 9.159 (p. 408) is non-additive. Therefore, Equation 9.155 (p. 407) and Equation 9.156 (p. 407) are now expressed as:

$$\begin{aligned} G_0^f&=-0.5K_fM_0^2f_{1/2}^{0,0}\\ G_r^f&=0.5K_fM_0^2\sum_{k=1}^{r-1}\binom{r}{k}f_{1/2}^{r,r-k};\quad r=2,3,4... \end{aligned} \tag{9.161}$$

where $f_l^{x,y}$ is the grid function defined as:

$$f_l^{x,y}=\sum_{i=1}^{\infty}\sum_{j=1}^{\infty}\left(i+j\right)^l i^x j^y\left(i^{1/3}+j^{1/3}\right)^2N_iN_j \tag{9.162}$$

Since the evaluation of this expression is not straightforward for l=1/2, the grid functions are evaluated for integer values of l. Then the grid functions for the fractional values are obtained using Lagrangian interpolation.

For example, for cases where three moments are solved, the following grid functions need to be calculated:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

- For r=0: $f_{1/2}^{0,0}$
- For r=1: $f_{1/2}^{1,0}$, $f_{1/2}^{1,1}$
- For r=2: $f_{1/2}^{2,0}$, $f_{1/2}^{2,1}$, $f_{1/2}^{1,2}$, $f_{1/2}^{2,2}$

These functions are obtained by first calculating $f_0^{x,y}$, $f_1^{x,y}$, and $f_2^{x,y}$ and then applying Lagrangian interpolation:

$$f_{1/2}^{x,y}=L\left(f_0^{x,y},f_1^{x,y},f_2^{x,y}\right)$$

Note that the grid function is symmetrical, that is $f_l^{x,y}=f_l^{y,x}$.

Coagulation in Transition Regime (0.1 < K_n < 10)

When the mean free path is of the order of the soot particle diameter, the coagulation term is calculated using the harmonic average from free molecular and continuum regimes in the following manner:

$$G_r^t=\frac{G_r^f G_r^c}{G_r^f+G_r^c} \tag{9.163}$$

9.2.2.4.5. Surface Growth and Oxidation

Once the soot is nucleated, carbon from the gas phase is deposited on the surface of soot particles, leading to a growth of soot particles' size as well as an increase in the soot mass. The soot formed due to nucleation is typically very small. Therefore, carbon deposition at the soot surface is the dominant mode of soot formation. In addition, the soot particles lose mass due to oxidation of the soot surface by O_2, OH, or any other oxidation species. The process of surface growth and oxidation are kinetically controlled and quite complex. Resolving the whole kinetics of these sub-processes involves large mechanisms and therefore computationally expensive. However, the process of soot surface growth and oxidation can be approximated by using a small mechanism (like a single step global mechanism) or reduced mechanisms.

One of the most widely used mechanism to model the surface growth and oxidation is the Hydrogen Abstraction C_2H_2 Addition (HACA) mechanism. The HACA mechanism represents the process of surface growth and oxidation by using reduced chemistry involving only few reaction steps.

There are a few variants of HACA mechanism. The HACA mechanism used in Ansys Fluent for modeling surface growth and oxidation are presented in Table 9.5: Arrhenius rate parameters for HACA mechanism (p. 410).

In Table 9.5: Arrhenius rate parameters for HACA mechanism (p. 410), the first six reactions are related to the surface growth and the last two reactions are related to oxidation due to O_2 and

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

OH, respectively. The reaction rates of these reactions have been taken from Appel et al. [22] (p. 1054).

Table 9.5: Arrhenius rate parameters for HACA mechanism

Reaction	$k=AT^{n}\exp(-E/RT)$			
	A (**cm^{3}mol^{-1}s^{-1}**)	n	E (**kcal/mol**)	k
$C_sH+H\rightarrow C_s^*+H_2$	4.2 x 10^{13}	0	13.0	k_1
$C_sH+H\leftarrow C_s^*+H_2$	3.9 x 10^{12}	0	11.0	k_1'
$C_sH+OH\rightarrow C_s^*+H_2O$	1.0 x 10^{10}	0.734	1.43	k_2
$C_sH+OH\leftarrow C_s^*+H_2O$	3.68 x 10^{8}	1.139	17.1	k_2'
$C_s^*+H\rightarrow C_sH$	2.0 x 10^{13}	0		k_3
$C_s^*+C_2H_2\rightarrow C_sH+H+2C(B)$	8.0 x 10^{7}	1.56	3.8	k_4
$C_s^*+O_2\rightarrow products$	2.2 x 10^{12}	0	7.5	k_5
$C_sH+OH\rightarrow CO+products$	Neoh at al. (1981) model, γ_{OH}=0.13			k_6

Since the oxidation by OH reaction (reaction 6 in Table 9.5: Arrhenius rate parameters for HACA mechanism (p. 410)) includes the collision between one gas-phase species and a surface species, the reaction rate constant is specified in terms of a sticking coefficient (reaction probability) γ_i. The unitless "Arrhenius-like" form of the sticking coefficient is defined as a probability where the computed values greater than one are set to the unity:

$$\gamma_i=\min\left(1,a_iT^{b_i}\exp\left(\frac{-c_i}{RT}\right)\right) \tag{9.164}$$

The sticking coefficient is converted to the mass-action kinetics rate constants using the collision frequency of the gas species with the bulk species:

$$k_{f_i}=\gamma_i\frac{\prod_{i=1}\sigma_j^{v'_{ji}}}{\Gamma_t^{m}}\sqrt{\frac{RT}{2\pi W_k}} \tag{9.165}$$

where

W_k = molecular weight of the gas-phase species

Γ_t = sum of total surface site concentration

m = sum of the stoichiometric coefficients of the reacting surface species

σ_j = occupancy of the surface species

v'_{ji} = reaction order of the surface species

In Equation 9.165 (p. 410), the term Γ_t^m accounts for the conversion of unitless sticking coefficient to appropriate units of the rate constant. The gas/surface collision frequency is considered by the term $\sqrt{\frac{RT}{2\pi W_k}}$. If the occupancy is unity for all species, then the production term $\prod_{i=1}\sigma_j^{v'_{ji}}$ is equal to one.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

The surface reactions are fast. The concentration of active radical sites are in steady state. The concentration of the active radical sites can be represented in terms of a fraction of the active sites:

$$\frac{C_s^*}{C_sH} = A = \frac{k_1[H] + k_2[OH] + k_6[OH]}{k_1'[H_2] + k_2'[H_2O] + k_3[H] + k_4[C_2H_2] + k_5[O_2]} \tag{9.166}$$

The concentration of active sites on the soot surface is χC_sH. Total active sites at soot surface are approximated as:

$$C_{\chi,soot} = \frac{\chi_s}{N_A}\alpha S \tag{9.167}$$

where

$\alpha = \tanh\left(a/\log\mu_1 + b\right)$ = fraction of active sites

a = 12.65 – 56.3e^{-4}T

b = -1.38 + 6.8e^{-4}T

χ_s = number of sites

S = surface area density (1/m)

The active radical sites are calculated as:

$$C_s^* = \alpha\chi_s\pi d_1^2\frac{A}{A+1} \tag{9.168}$$

where d_1 is the carbon atom diameter.

The moment source terms due to the surface growth reaction and oxidation are calculated as follows:

- Surface growth due to an addition of C_2H_2:

$$W_r^{C_2H_2} = k_4[C_s^*][C_2H_2]M_0\sum_{k=0}^{r-1}\binom{r}{k}(\Delta)^{r-k}\mu_{k+2/3} \tag{9.169}$$

- Oxidation due to the presence of O_2:

$$W_r^{O2} = k_5[C_s^*][O_2]M_0\sum_{k=0}^{r-1}\binom{r}{k}(-\Delta)^{r-k}\mu_{k+2/3} \tag{9.170}$$

- Surface growth due to deposition of OH:

$$W_r^{OH} = k_6[C_SH][OH]M_0\sum_{k=0}^{r-1}\binom{r}{k}(-\Delta)^{r-k}\mu_{k+2/3} \tag{9.171}$$

In Equation 9.169 (p. 411) through Equation 9.171 (p. 411), Δ is the number of carbon atoms removed or added:

- Δ = 2 for surface growth and oxidation due to O_2

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

- Δ = 1 for surface growth and oxidation due to OH

For the r-th moment, the total source term due to surface growth and oxidation is

$$W_r = W_r^{C_2H_2} + W_r^{O_2} + W_r^{OH} \tag{9.172}$$

Note that with the HACA mechanism, acetylene is consumed during the surface growth of soot. Since Ansys Fluent generates the flamelets before the solution, the flamelets have no knowledge of the soot process, which can cause an over-prediction of the soot surface growth and eventually higher soot yields with non-premixed/partially premixed combustion models.

9.2.2.4.6. Soot Aggregation

In Coagulation (p. 406), soot particle coagulation assumes coalescent soot growth, whereby the larger particle resulting from collision of two smaller particles is always spherical. However, in many practical combustion devices, the growth of soot may not be limited to the coalescent regime. The colliding particles can stick to each other and form chain-like structures called aggregates. The aggregate has a larger surface area than a coalesced single sphere of the same mass. The formation of the aggregates, therefore, enhances the rate of coagulation process. Ignoring the aggregate formation could lead to errors in calculations of the coagulation and the soot surface growth rates.

In the regime of aggregate collision, the soot particle size distribution becomes bivariate. The number of primary particles within the aggregate represents the internal coordinate, while the aggregate mass and size represents the external coordinate. Resolving a two-dimensional PDF is not trivial and computationally expensive. Ansys Fluent has adopted the approach proposed by Kazakov and Frenklach [297] (p. 1070) in which the method of moments for modeling the soot formation is extended to account for soot aggregation. The current formulation separates the two coagulation regimes, namely, the coalescence and aggregation, switching between the two based on a user-specified critical diameter. The coagulation is assumed to be coalescent if the average particle diameter is less than the critical diameter. When the soot particle diameter reaches the critical value, the coagulation is automatically switched to the aggregation regime.

In the presence of the aggregation process, Equation 9.136 (p. 402) through Equation 9.139 (p. 403) represent the size distribution and the transport of the soot aggregate mass (that is, the concentration moments). In addition to the aggregate moments, similar to Equation 9.136 (p. 402), the discrete form of concentration moment for the distribution of the primary particles in the aggregates is defined as:

$$^{P}M_r = \sum_{i=1}^{\infty} n_i^r N_i \tag{9.173}$$

where $^{P}M_r$ is the particle r^{th} moment, and n_i^r is the number of the primary particles in i^{th} class. Similar to the aggregate mass (or global) moments, here also the concept of the reduced particle moment can be used:

$$\theta_r = \frac{^{P}M_r}{^{P}M_0} \tag{9.174}$$

where θ_r is the r^{th} reduced moment of the primary particles within the aggregate.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

By definitions in Equation 9.136 (p. 402) and Equation 9.173 (p. 412), the particle zeroth moment ${}^{P}M_0$ and the global zeroth moment M_0 are identical, that is:

$$ {}^{P}M_0 = M_0 \tag{9.175} $$

Also, the average number of the particles in an aggregate, n_p, is given by the first two moments as follows:

$$ \left\langle n_p \right\rangle = \theta_1 = \frac{{}^{P}M_1}{{}^{P}M_0} \tag{9.176} $$

Similar to the aggregate moments, particle moments will also evolve due to the various soot formation sub-steps, such as, nucleation and coagulation, as well as particle transport in the physical space. The governing equations for the change in the particle moments can be written as:

$$ \frac{d^{P}M_r}{dt} = R_0 + H_r, \qquad r = 1,2,3... \tag{9.177} $$

where,

R_0 = the nucleation source term calculated by Equation 9.140 (p. 404)

H_r = source due to the coagulation of the aggregates

Because Equation 9.175 (p. 413) already yields the solution for $r = 0$, Equation 9.177 (p. 413)) is only solved for $r > 0$.

The first particle moment ${}^{P}M_1$ gives the average number of the primary particles n_p (Equation 9.176 (p. 413)) from which the particle mass can be calculated. $H_1 = 0$ because the particles total mass does not change due to coagulation process.

For higher moments, H_r in equation Equation 9.177 (p. 413) needs to be computed. Using Equation 9.148 (p. 406) along with the definition of particle moments in Equation 9.173 (p. 412), H_r can be written as

$$ H_r = 0.5 \sum_{k=1}^{r-1} \binom{r}{k} \sum_{i=1}^{\infty} \sum_{j=1}^{\infty} n_i^k n_j^{r-k} \beta_{i,j} N_i N_j, \qquad for\ r \geq 2 \tag{9.178} $$

where β is the aggregate collision coefficient. Its value is dependent on the collision regime, which is calculated using the Knudsen number K_n (Equation 9.149 (p. 406)).

Aggregate Coagulation in Continuum Regime ($K_n < 0.1$)

The aggregate collision coefficient in the continuum regime is expressed as:

$$ {}^{a}\beta_{i,j} = {}^{a}\beta^{c}_{i,j} = K_C \left(\frac{C_i}{d_i} + \frac{C_j}{d_j} \right) \left(d_i + d_j \right) \tag{9.179} $$

where d_i and d_j are the collision diameters of the aggregates of different size classes i and j, respectively. The rest of notation is the same as in equation Equation 9.153 (p. 407). For coalescent collision, the collision diameter and the particle diameter are assumed to be the same. However, due to aggregation, the collision diameter can differ significantly from the particle diameter depending on the fractal structure of the aggregate. An aggregate is composed of a number of

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

primary particles, which are assumed to be spherical and of equal size. The collision diameter of the aggregate is calculated from the following relation:

$$d = d_p n_p^{1/F} \tag{9.180}$$

where d_p is the diameter of the primary particle, and F is the fractal dimension that describes the fractal structure of the aggregate. A value of F between 1.7 and 2.0 is found to be reasonably good and suggested in the literature for aggregate coagulation. In the pure coalescent coagulation, $F = 3$. The mass of the aggregate can be specified in the following manner:

$$m = n_p\left(\frac{\pi}{6}\rho d_p^3\right) \tag{9.181}$$

From Equation 9.180 (p. 414) and Equation 9.181 (p. 414), the aggregate collision diameter can be written as:

$$d = \left(\frac{6}{\rho\pi}m\right)^{1/3} n_p^{(1/F-1/3)} \tag{9.182}$$

Substituting the value of the collision diameter from Equation 9.182 (p. 414) into Equation 9.179 (p. 413) and replacing the value of the collision coefficient in Equation 9.178 (p. 413), we obtain the aggregate coagulation source term for the r^{th} moment:

$$H_r^c = 0.5K_c\sum_{k=1}^{r-1}\binom{r}{k}\left[\begin{array}{l} 2\theta_k\theta_{r-k} + \left\langle m_i^{1/3}n_i^{k+\varphi}\right\rangle\left\langle m_j^{-1/3}n_j^{r-k-\varphi}\right\rangle + \left\langle m_i^{-1/3}n_i^{k-\varphi}\right\rangle\left\langle m_j^{1/3}n_j^{r-k+\varphi}\right\rangle \\ +K_c'\left\{\begin{array}{l}\left\langle m_i^{-1/3}n_i^{k-\varphi}\right\rangle\theta_{r-k} + \left\langle m_j^{-1/3}n_i^{r-k-\varphi}\right\rangle\theta_k + \left\langle m_i^{1/3}n_i^{k+\varphi}\right\rangle\left\langle m_j^{-2/3}n_j^{r-k-2\varphi}\right\rangle + \\ \left\langle m_i^{-2/3}n_i^{k-2\varphi}\right\rangle\left\langle m_j^{1/3}n_j^{r-k+\varphi}\right\rangle\end{array}\right\}\end{array}\right]M_0^2 \tag{9.183}$$

where,

$$\varphi = \frac{1}{F} - \frac{1}{3} \tag{9.184}$$

Equation 9.183 (p. 414) involves binary moments of the soot aggregate size and the number of primary particles. The calculations for binary moments are not trivial and require a multi-dimensional PDF of the particle size distribution. Kazakov and Frenklach [297] (p. 1070) suggested approximating the binary moment using two one-dimensional moments as follows:

$$\left\langle m^x n^y\right\rangle \approx \left\langle m^x\right\rangle\left\langle n^y\right\rangle = \mu_x\theta_y \tag{9.185}$$

Thus, Equation 9.183 (p. 414) can be written as:

$$H_r^c = 0.5K_c\sum_{k=1}^{r-1}\binom{r}{k}\left[\begin{array}{l} 2\theta_k\theta_{r-k} + \mu_{1/3}\theta_{k+\varphi}\mu_{-1/3}\theta_{r-k-\varphi} + \mu_{-1/3}\theta_{k+\varphi}\mu_{1/3}\theta_{r-k+\varphi} \\ +K_c'\left\{\begin{array}{l}\mu_{-1/3}\theta_{k-\varphi}\theta_{r-k} + \mu_{-1/3}\theta_{r-k-\varphi}\theta_k + \mu_{1/3}\theta_{k+\varphi}\mu_{-2/3}\theta_{r-k-2\varphi} + \\ \mu_{-2/3}\theta_{k-2\varphi}\mu_{1/3}\theta_{r-k+\varphi}\end{array}\right\}\end{array}\right]M_0^2 \tag{9.186}$$

Equation 9.186 (p. 414) contains the fractional and intermediate primary particle moments, which are calculated using the same interpolation method employed to aggregate moments (see Equation 9.157 (p. 408) and Equation 9.158 (p. 408)).

Aggregate Coagulation in Free Molecular Regime ($K_n > 10$)

The collision coefficient in the free molecular regime can be written as:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$ {}^{a}\beta_{i,j}^{f} = K_f \sqrt{\frac{m_i + m_j}{m_i m_j}} \left(d_i + d_j \right)^2 \tag{9.187} $$

where m and d are the mass and collision diameter of the aggregate, respectively. The collision diameter is calculated from Equation 9.182 (p. 414). Similar to the coalescent coagulation in the free molecular regime calculations, difficulties arise when calculating the source due to the coagulation of the aggregates H_r (Equation 9.178 (p. 413)) because of the non-additive collision coefficient term from Equation 9.187 (p. 415). In a manner similar to Equation 9.161 (p. 408), H_r can be expressed as:

$$ H_r^f = 0.5 K_f M_0^2 \sum_{k=1}^{r-1} \binom{r}{k} h_{1/2}^{r,r-k}; \qquad r = 2, 3, 4.... \tag{9.188} $$

where $h_{1/2}$ is the aggregate grid function. Equating the right-hand sides of Equation 9.178 (p. 413) and Equation 9.188 (p. 415) and substituting the collision coefficient from equation Equation 9.187 (p. 415), the aggregate grid function can be written as:

$$ h_l^{x,y} = \frac{1}{M_0^2} \left(\sum_{q=0}^{l} \binom{l}{q} \left(\sum_{i=1}^{\infty} \sum_{j=1}^{\infty} m_i^q m_j^{l-q} m_i^{1/2} m_j^{1/2} \left(d_i^{1/3} + d_j^{1/3} \right)^2 n_i^x n_j^y N_i N_j \right) \right) \tag{9.189} $$

The collision diameter in Equation 9.189 (p. 415) is given by equation Equation 9.182 (p. 414). Therefore, Equation 9.189 (p. 415) can be rewritten as

$$ h_l^{x,y} = \frac{1}{M_0^2} \sum_{q=0}^{l} \binom{l}{q} \sum_{i=1}^{\infty} \sum_{j=1}^{\infty} \begin{pmatrix} \left\langle m_i^{q+1/6} n_i^{x+2\varphi} \right\rangle \left\langle m_j^{l-q-1/2} n_j^{y} \right\rangle + \left\langle m_i^{q-1/2} n_i^{x} \right\rangle \left\langle m_j^{l-q+1/6} n_j^{y+2\varphi} \right\rangle \\ + 2 \left\langle m_i^{q-1/6} n_i^{x+\varphi} \right\rangle \left\langle m_j^{l-q-1/6} n_j^{y+\varphi} \right\rangle N_i N_j \end{pmatrix} \tag{9.190} $$

Using Equation 9.136 (p. 402), Equation 9.173 (p. 412), and Equation 9.185 (p. 414), the aggregate grid function can be expressed as follows:

$$ h_l^{x,y} = \sum_{q=0}^{l} \binom{l}{q} \left(\mu_{q+1/6} \theta_{x+2\varphi} \mu_{l-q-1/2} \theta_y + \mu_{q-1/2} \theta_x \mu_{l-q+1/6} \theta_{y+2\varphi} + 2 \mu_{q-1/6} \theta_{x+\varphi} \mu_{l-q-1/6} \theta_{y+\varphi} \right) \tag{9.191} $$

Aggregate Coagulation in Transition Regime ($0.1 < K_n < 10$)

Similar to Equation 9.163 (p. 409), the coagulation term in the transition regime is calculated using the harmonic average from the free molecular and the continuum regimes.

The coagulation and surface growth rate source terms for global moments are calculated using Equation 9.155 (p. 407), Equation 9.156 (p. 407), Equation 9.161 (p. 408), and Equation 9.172 (p. 412). These source terms are dependent on the diameter and the surface area of the coagulated particle. When the aggregation model is enabled, the resulting diameter and surface area of the aggregate will be larger than those of the spherical particle. The coagulation and surface growth rate terms for aggregate moments are modified as further described.

Modified coagulation term

The modified continuum regime equation Equation 9.155 (p. 407) that includes the change in collision diameter, is follows:

$$ G_0^c = -K_c \left[1 + \mu_{1/3} \theta_{\varphi} \mu_{-1/3} \theta_{-\varphi} + K_c' \left(\mu_{-1/3} \theta_{-\varphi} + \mu_{1/3} \theta_{\varphi} \mu_{-2/3} \theta_{2\varphi} \right] M_0^2 \right. \tag{9.192} $$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Similarly, Equation 9.156 (p. 407), which is the coagulation source for r^{th} moment, is calculated as:

$$G_r^c = 0.5K_c \sum_{k=1}^{r-1} \binom{r}{k} \begin{bmatrix} 2\mu_k \mu_{r-k} + \mu_{k+1/3}\theta_{\varphi}\mu_{r-k-1/3}\theta_{-\varphi} + \mu_{k-1/3}\theta_{-\varphi}\mu_{r-k+1/3}\theta_{\varphi} + K_c' \\ (\mu_{k-1/3}\theta_{-\varphi}\mu_{r-k} + \mu_k \mu_{r-k-1/3}\theta_{-\varphi} + \mu_{k+1/3}\theta_{\varphi}\mu_{r-k-2/3}\theta_{-2\varphi} + \mu_{k-2/3}\theta_{-2\varphi}\mu_{r-k+1/3}\theta_{\varphi} \end{bmatrix} M_0^2 \quad (9.193)$$

In the free molecular regime, the grid function $f_l^{x,y}$ in Equation 9.162 (p. 408) is modified to account for aggregation as follows:

$$f_l^{x,y} = \frac{1}{M_0^2} \sum_{i=1}^{\infty} \sum_{j=1}^{\infty} (m_i + m_j)^l m_i^{x-1/2} m_j^{y-1/2} \left(m_i^{1/3} n_i^{\varphi} + m_j^{1/3} n_j^{\varphi}\right)^2 N_i N_j \quad (9.194)$$

Equation 9.194 (p. 416) can be written in the alternative form of reduced moments as follows:

$$f_l^{x,y} = \sum_{k=0}^{l} \binom{l}{k} \left(\mu_{k+x+1/6}\theta_{2\varphi}\mu_{l-k+y-1/2} + \mu_{k+x-1/2}\mu_{l-k+y+1/6}\theta_{2\varphi} + 2\mu_{k+x-1/6}\theta_{\varphi}\mu_{l-k+y-1/6}\theta_{\varphi}\right) \quad (9.195)$$

Equation 9.192 (p. 415) through Equation 9.195 (p. 416) are obtained by substituting the modified value of the collision diameter from Equation 9.182 (p. 414) into Equation 9.155 (p. 407), Equation 9.156 (p. 407), and Equation 9.161 (p. 408).

Modified surface growth and the oxidation rate terms

Similarly, the surface growth and the oxidation rate for different moments are enhanced by a factor $\theta_{1/3}$ and calculated as follows:

$$\begin{aligned} W_{,r,AGG}^{C2H2} &= \theta_{1/3} W_{,r,NO_AGG}^{C2H2} \\ W_{,r,AGG}^{O2} &= \theta_{1/3} W_{,r,NO_AGG}^{O2} \\ W_{,r,AGG}^{OH} &= \theta_{1/3} W_{,r,NO_AGG}^{OH} \end{aligned} \quad (9.196)$$

9.3. Decoupled Detailed Chemistry Model

The following sections present background information about the detailed chemistry pollutant model. To learn how to use this model, see Using the Decoupled Detailed Chemistry Model in the User's Guide.

9.3.1. Overview

9.3.2. Decoupled Detailed Chemistry Model Theory

9.3.1. Overview

Many pollutant species, such as NO_x, are slowly forming relative to the major combustion products, and require a finite-rate chemistry calculation. Further, the mass fractions of these pollutant species are usually very small and they have negligible influence on the flow fields. Hence, for steady-state simulations, the solution of the pollutant transport equations can be decoupled from the other equations and solved in a postprocessing step. Ansys Fluent offers several specific pollutant models (NO_x, soot) with fixed chemical mechanisms. The detailed chemistry pollutant model allows pollutant postprocessing using any chemical kinetic mechanism.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

9.3.1.1. Limitations

The following limitations exist when using the detailed chemistry pollutant model:

- The detailed chemistry pollutant model is only available with the steady-state solver.
- Since pollutant chemistry is postprocessed on a steady-state solution, pollutant species cannot couple with the flow, energy, or any other field (for example, radiating soot).
- It can be difficult to revert to the original settings of your case file after you have set up the decoupled detailed chemistry pollutant model. This limitation can be addressed by saving case and data files prior to setting up the decoupled detailed chemistry pollutant model.

9.3.2. Decoupled Detailed Chemistry Model Theory

A decoupled detailed chemistry pollutant calculation starts from a converged steady-state combustion solution. This original solution can be obtained with the Species Transport, Non-Premixed, Partially-Premixed, or PDF Transport combustion models. After you have imported a detailed chemical mechanism through the CHEMKIN import facility, you will identify the pollutant species. The non-pollutant species in the detailed chemical mechanism that are not already in the original combustion solution are determined by a chemical equilibrium calculation at the fixed temperature of the original solution. Note that species that are not identified as pollutants and do not participate in the reactions amongst the pollutant species are eliminated. Similarly, reactions in the imported CHEMKIN mechanism that do not include any pollutant species are also eliminated.

The sequence of steps that occurs after the pollutant mechanism has been imported and the pollutant species selected is as follows:

1. A mixture material called **pollutant-mixture** is created and set as the default.
2. Non-pollutant, non-original species in the imported mechanism are set to their chemical equilibrium values at the cell temperature.
3. The Species Transport combustion model is enabled with the Laminar sub-model and the stiff chemistry solver.
4. The solution to all transport equations, except the selected pollutant species, is disabled.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information
of ANSYS, Inc. and its subsidiaries and affiliates.

Chapter 10: Engine Ignition

This chapter discusses the theory behind the engine ignition models available in Ansys Fluent. Information can be found in the following sections.

10.1. Spark Model

10.2. Autoignition Models

10.3. Crevice Model

For information about using these ignition models, see Modeling Engine Ignition in the User's Guide.

10.1. Spark Model

The spark model in Ansys Fluent will be described in the context of the premixed and partially premixed combustion models, including the species transport model. For information about using this model, see Using the Spark Model in the User's Guide. Information regarding the theory behind this model is detailed in the following sections:

10.1.1. Overview and Limitations

10.1.2. Spark Model Theory

10.1.1. Overview and Limitations

Initiation of combustion at a desired time and location in a combustion chamber can be accomplished by sending a high voltage across two narrowly separated wires, creating a spark. The spark event in typical engines happens very quickly relative to the main combustion in the engine. The physical description of this simple event is very involved and complex, making it difficult to accurately model the spark in the context of a multidimensional engine simulation. Additionally, the energy from the spark event is several orders of magnitude less than the chemical energy release from the fuel. Despite the amount of research devoted to spark ignition physics and ignition devices, the ignition of a mixture at a point in the domain is more dependent on the local composition than on the spark energy (see Heywood [241] (p. 1067)). Thus, for situations in which Ansys Fluent is utilized for combustion engine modeling, including internal combustion engines, the spark event does not need to be modeled in great detail, but simply as the initiation of combustion over a duration, which you will set.

Since spark ignition is inherently transient, the spark model is only available in the transient solver. Additionally, the spark model requires chemical reactions to be solved. The spark model is available for the species transport and all the turbulent combustion models, except the non-premixed model.

10.1.2. Spark Model Theory

Typically, the initial spark size is small relative to the cell size, and the spark is under-resolved on the CFD mesh. Modeling the spark by burning a few cells around the spark location shows strong sensitivity to the grid and time-step size, and flame speed and flame brush diffusion can be erroneous due

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

to the insufficient space and time resolution. In addition, for cases where the initial spark is smaller than the cell size, ignition proceeds too quickly.

To mitigate this sensitivity, Ansys Fluent solves a sub-grid equation for the spark evolution. The spark shape is assumed to be perfectly spherical, with an infinitely thin flame front. The spark radius, r, grows in time, t, according to the ODE,

$$\frac{dr}{dt}=\frac{\rho_u}{\rho_b}S_t \tag{10.1}$$

where ρ_u is the density of the unburned fluid ahead of the flame front, ρ_b is the density of the burnt fluid behind of the flame, and S_t is the turbulent flame speed.

The sub-grid spark model is transferred to the CFD grid through a representative volume of CFD cells. This volume is spherical and has a fixed diameter computed as the local turbulent length scale at the spark location. However, in order to ensure that this representative volume is not too large (relative to the size of the combustor) or too small (relative to the cell size), the radius of the representative sphere, r_t, is calculated as,

$$r_t=\max\left(r_o+3\Delta,\, 3r_o,\, \min\left(\frac{1}{2}l_t, r_o+10\Delta\right)\right) \tag{10.2}$$

where r_o is the user-specified initial spark radius, Δ is the cell length scale, and l_t is the turbulent length scale.

Alternatively r_t may be specified as a fixed value via the spark-model text interface.

l_t is taken to define the representative volume because, once the spark diameter reaches this size, the flame speed is affected by all the turbulent scales present. The spark radius will continue to increase until the length scale is reached, even if the simulation time exceeds the user-specified duration. Note, that the specified duration is only used to calculate the rate of spark energy input and the time when this input ceases. At this condition Ansys Fluent automatically switches the spark flame speed model off, so that the flame speed is modeled using the flame speed model that you have selected in the **Species** dialog box, throughout the domain.

For the G-Equation model, reaction progress in the spark is calculated in the usual way for this model as given by Equation 8.76 (p. 336).

The temperature and species composition (denoted by φ) within the representative spherical volume at every time-step are calculated as,

$$\varphi=c\varphi^b+(1-c)\varphi^u \tag{10.3}$$

where φ^b denotes the equilibrium burnt composition, and φ^u is the unburned composition. These compositions are fixed in time and uniform in space.

Since the thermo-chemical state behind the spark flame front is instantaneously equilibrated as the spark propagates, spark energy is not required to ignite the mixture. By default, the spark energy is set to zero for all combustion models, and the burnt temperature is the equilibrium temperature. However, spark energy can be set to a positive value in the user interface, in which case the temperature behind the spark will be higher than the equilibrium temperature.

Ansys Fluent offers the following models for the turbulent flame speed, S_t in Equation 10.1 (p. 420):

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Turbulent Curvature

In the Turbulent Curvature model, the turbulent flame speed is calculated as,

$$S_t = \max\left(S_l - \frac{2D}{r}, S_t(r) - \frac{2D_t}{r}\right) \tag{10.4}$$

where S_l is the laminar flame speed, r is the current spark radius, D is the laminar diffusivity, and D_t is the turbulent diffusivity. D and D_t are evaluated at the spark location, and $S_t(r)$ is the turbulent flame speed evaluated at the turbulent length scale of the spark radius. Since turbulent scales larger than the spark radius convect the spark but do not increase its area and flame speed, only turbulent length scales up to the spark radius r can affect the turbulent flame speed of the spark. For the pre-mixed and partially-premixed models, the spark flame speed model is the same flame speed model as selected for the main combustion. For Species Transport cases, by default the Zimont flame speed model is used. S_l and D are added as additional material inputs in the interface for the species transport models.

Note that the effect of the flame curvature is to decrease both the laminar and turbulent flames speeds. Since the initial spark radius r_o is a user-input, decreasing r_o slows the spark propagation and increases the burning time.

Turbulent Length

In the Turbulent Length model, the turbulent flame speed is calculated as,

$$S_t = \max\left(S_l, S_t(r)\right) \tag{10.5}$$

That is, the Turbulent Length model ignores the effects of flame curvature on the flame speed.

Herweg-Maly

The turbulent flame speed is calculated using the model proposed by Herweg and Maly [242] (p. 1067):

$$S_t = \max\left(S_l, S_l\left(I_0 + I_0^{\frac{1}{2}}\left(\frac{u'}{u'+S_l}\right)^{\frac{1}{2}}\left[1 - exp\left(-\frac{r}{l_t}\right)\right]^{\frac{1}{2}}\left[1 - exp\left(-\frac{t-t_0}{\tau_s}\right)\right]^{\frac{1}{2}}\left(\frac{u'}{S_l}\right)^{\frac{5}{6}}\right)\right) \tag{10.6}$$

where I_0 is a function for effect of strain on the laminar burning velocity calculated as,

$$I_0 = \max\left(0,\ 1 - \left(\frac{\delta}{15 l_t}\right)^{\frac{1}{2}}\left(\frac{u'}{S_l}\right)^{\frac{3}{2}} - 2\frac{\delta}{r}\frac{\rho_u}{\rho_b}\right)$$

and

t = current time

t_0 = start time

u' = turbulent velocity scale

$\tau_s = \frac{l_t}{u'+S_l}$

δ = laminar flame thickness = $\frac{D}{S_l}$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

If the Blint modifier is being used

$$\delta = \text{laminar flame thickness} = \frac{D}{S_l} B_1 \left(\frac{\rho_u}{\rho_b} \right)^{B_2}$$

where

B_1 = 2.0

B_2 = 0.7

ρ_u is the unburned density

ρ_b is the burned density

Laminar

In the Laminar model, the turbulent flame speed S_t in Equation 10.1 (p. 420) is modeled as the laminar flame speed, S_l. Since the Laminar flame speed can be modeled with a user-defined function (UDF), the Laminar option can be used to define your own function for the turbulent flame speed.

10.2. Autoignition Models

Autoignition phenomena in engines are due to the effects of chemical kinetics of the reacting flow inside the cylinder. There are two types of autoignition models considered in Ansys Fluent:

- knock model in spark-ignited (SI) engines
- ignition delay model in diesel engines

For information regarding using autoignition models, see Using the Autoignition Models in the User's Guide. The theory behind the autoignition models is described in the following sections:

10.2.1. Model Overview

10.2.2. Model Limitations

10.2.3. Ignition Model Theory

10.2.1. Model Overview

The concept of knock has been studied extensively in the context of premixed engines, as it defines a limit in terms of efficiency and power production of that type of engine. As the compression ratio increases, the efficiency of the engine as a function of the work extracted from the fuel increases.

However, as the compression ratio increases, the temperature and pressure of the air/fuel mixture in the cylinder also increase during the cycle compressions. The temperature and pressure increase can be large enough for the mixture to spontaneously ignite and release its heat before the spark plug fires. The premature release of all of the energy in the air/fuel charge is almost never desirable, as this results in the spark event no longer controlling the combustion. As a result of the premature release of the energy, catastrophic damage to the engine components can occur. The sudden, sharp rise in pressure inside the engine can be heard clearly through the engine block as a knocking sound, hence

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

the term "knock". For commonly available gasoline pumps, knock usually limits the highest practical compression ratio to less than 11:1 for premium fuels and around 9:1 for less expensive fuels.

By comparison, ignition delay in diesel engines has not been as extensively studied as SI engines, mainly because it does not have such a sharply defining impact on engine efficiency. Ignition delay in diesel engines refers to the time between when the fuel is injected into the combustion chamber and when the pressure starts to increase as the fuel releases its energy. The fuel is injected into a gas that is usually air, however, it can have a considerable amount of exhaust gas mixed in (or EGR) to reduce nitrogen oxide emissions (NOx). Ignition delay depends on the composition of the gas in the cylinder, the temperature of the gas, the turbulence level, and other factors. Since ignition delay changes the combustion phasing, which in turn impacts efficiency and emissions, it is important to account for it in a diesel engine simulation.

10.2.2. Model Limitations

The main difference between the knock model and the ignition delay model is the manner in which the model is coupled with the chemistry. The knock model always releases energy from the fuel while the ignition delay model prevents energy from being released prematurely.

The knock model in Ansys Fluent is compatible with the premixed and partially premixed combustion models. The autoignition model is compatible with any volumetric combustion model, with the exception of the purely premixed models. The autoignition models are inherently transient and so are not available with steady simulations.

The autoignition models in general require adjustment of parameters to reproduce engine data and are likely to require tuning to improve accuracy. Once the model is calibrated to a particular engine configuration, then different engine speeds and loads can be reasonably well represented. Detailed chemical kinetics may be more applicable over a wider range of conditions, though are more expensive to solve. The single equation autoignition models are appropriate for the situation where geometric fidelity or resolution of particular flow details is more important than chemical effects on the simulation.

10.2.3. Ignition Model Theory

Both the knock and the ignition delay models are treated similarly in Ansys Fluent, in that they share the same infrastructure. These models belong to the family of *single equation* autoignition models and use correlations to account for complex chemical kinetics. They differ from the eight step reaction models, such as Halstead's "Shell" model [228] (p. 1066), in that only a single transport equation is solved. The source term in the transport equation is typically not stiff, therefore making the equation relatively inexpensive to solve.

This approach is appropriate for large simulations where geometric accuracy is more important than fully resolved chemical kinetics. The model can be used on less resolved meshes to explore a range of designs quickly, and to obtain trends before utilizing more expensive and presumably more accurate chemical mechanisms in multidimensional simulations.

10.2.3.1. Transport of Ignition Species

Autoignition is modeled using the transport equation for an *Ignition Species*, Y_{ig}, which is given by

$$\frac{\partial \rho Y_{ig}}{\partial t} + \nabla \cdot \left(\rho \vec{v} Y_{ig} \right) = \nabla \cdot \left(\frac{\mu_t}{Sc_t} \nabla Y_{ig} \right) + \rho S_{ig} \tag{10.7}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

where Y_{ig} is a "mass fraction" of a passive species representing radicals that form when the fuel in the domain breaks down. Sc_t is the turbulent Schmidt number. The term S_{ig} is the source term for the ignition species, which has a form,

$$S_{ig} = \int_{t=t_0}^{t} \frac{dt}{\tau_{ig}}$$

where t_0 corresponds to the time at which fuel is introduced into the domain. The τ_{ig} term is a correlation of ignition delay with the units of time. Ignition has occurred when the ignition species reaches a value of 1 in the domain. It is assumed that all the radical species represented by Y_{ig} diffuse at the same rate as the mean flow.

Note that the source term for these radical species is treated differently for knock and ignition delay. Furthermore, the form of the correlation of ignition delay differs between the two models. Details of how the source term is treated are covered in the following sections.

10.2.3.2. Knock Modeling

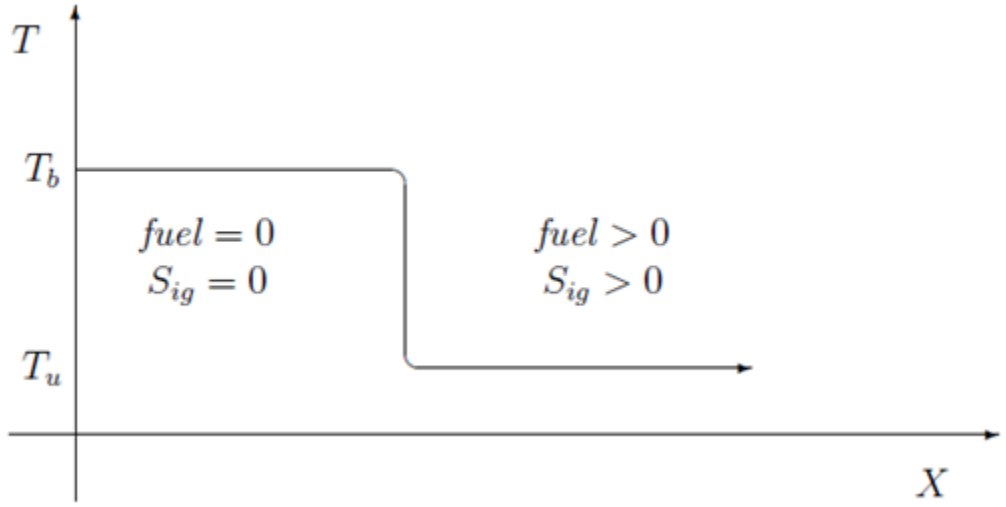
When modeling knock or ignition delay, chemical energy in the fuel is released when the ignition species reaches a value of 1 in the domain. For the knock model, two correlations are built into Ansys Fluent. One is given by Douaud [147] (p. 1061), while the other is a generalized model that reproduces several correlations, given by Heywood [241] (p. 1067).

10.2.3.2.1. Modeling of the Source Term

In order to model knock in a physically realistic manner, the source term is accumulated under appropriate conditions in a cell. Consider the one dimensional flame in Figure 10.1: Flame Front Showing Accumulation of Source Terms for the Knock Model (p. 425). Here, the flame is propagating from left to right, and the temperature is relatively low in front of the flame and high behind the flame. In this figure, T_b and T_u represent the temperatures at the burned and unburned states, respectively. The ignition species accumulates only when there is fuel. In the premixed model, the fuel is defined as $fuel = 1 - c$, where c is the progress variable. If the progress variable has a value of zero, the mixture is considered unburned. If the progress variable is 1, then the mixture is considered burned.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Figure 10.1: Flame Front Showing Accumulation of Source Terms for the Knock Model



When the ignition species reaches a value of 1 in the domain, knock has occurred at that point. The value of the ignition species can exceed unity. In fact, values well above that can be obtained in a short time. The ignition species will continue to accumulate until there is no more fuel present.

10.2.3.2.2. Correlations

An extensively tested correlation for knock in SI engines is given by Douaud and Eyzat [147] (p. 1061):

$$\tau = 0.01768\left(\frac{ON}{100}\right)^{3.402} p^{-1.7}\exp\left(\frac{3800}{T}\right) \tag{10.8}$$

where ON is the octane number of the fuel, p is the absolute pressure in atmospheres and T is the temperature in Kelvin.

A generalized expression for τ is also available that can reproduce many existing Arrhenius correlations. The form of the correlation is

$$\tau = A\left(\frac{ON}{100}\right)^{a} p^{b} T^{c} \mathrm{RPM}^{d} \Phi^{d} \exp\left(\frac{-E_a}{RT}\right) \tag{10.9}$$

where A is the pre-exponential (with units in seconds), RPM is the engine speed in cycles per minute and Φ is the fuel/air equivalence ratio.

10.2.3.2.3. Energy Release

Once ignition has occurred in the domain, the knock event is modeled by releasing the remaining fuel energy with a single-step Arrhenius reaction. An additional source term, which burns the remaining fuel in that cell, is added to the rate term in the premixed model. The reaction rate is given by

$$\dot{\omega} = A_0 \exp\frac{-E_a}{RT} \tag{10.10}$$

where A_0 = 8.6×10^{11}, and E_a/R = 15078 These values are chosen to reflect single-step reaction rates appropriate for propane as described in Amsden [16] (p. 1053). The rate at which the fuel is

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

consumed is limited such that a completely unburned cell will burn during three of the current time steps. Limiting the reaction rate is done purely for numerical stability.

10.2.3.3. Ignition Delay Modeling

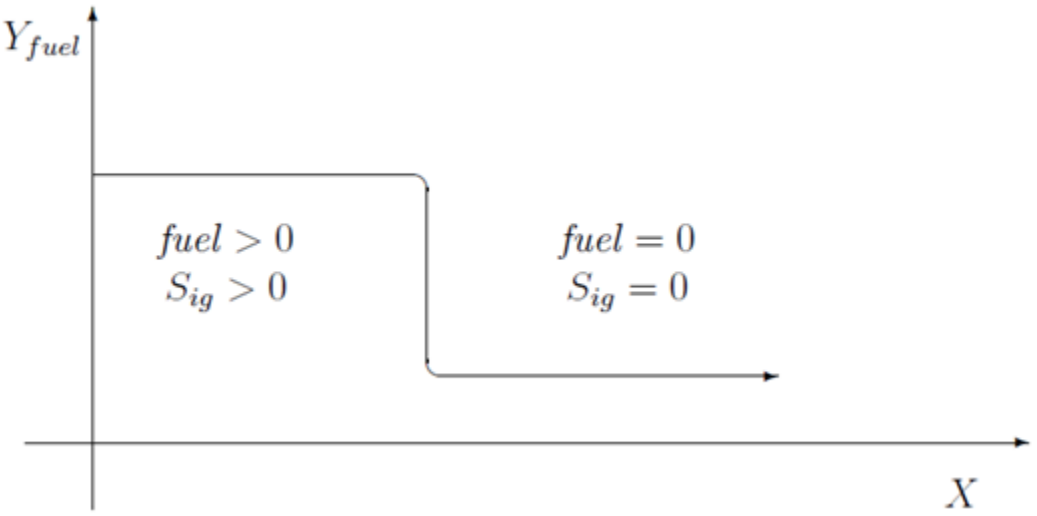
When modeling ignition delay in diesel engines, chemical reactions are allowed to occur when the ignition species reaches a value of 1 in the domain. For the ignition delay model, two correlations are built into Ansys Fluent, one given by Hardenburg and Hase [234] (p. 1066) and the other, a generalized model that reproduces several Arrhenius correlations from the literature.

If the ignition species is less than 1 when using the ignition delay model, the chemical source term is suppressed by not activating the combustion model at that particular time step; therefore, the energy release is delayed. This approach is reasonable if you have a good high-temperature chemical model, but do not want to solve for typically expensive low temperature chemistry.

10.2.3.3.1. Modeling of the Source Term

In order to model ignition in a physically realistic manner, the source term is accumulated under appropriate conditions in a cell. Consider the one dimensional spray in Figure 10.2: Propagating Fuel Cloud Showing Accumulation of Source Terms for the Ignition Delay Model (p. 426).

Figure 10.2: Propagating Fuel Cloud Showing Accumulation of Source Terms for the Ignition Delay Model



Here, the spray is propagating from left to right and the fuel mass fraction is relatively low in front of the spray and high behind the spray. If there is no fuel in the cell, the model will set the local source term to zero, nevertheless, the value of Y_{ig} can be nonzero due to convection and diffusion.

10.2.3.3.2. Correlations

If fuel is present in the cell, there are two built-in options in Ansys Fluent to calculate the local source term. The first correlation was done by Hardenburg and Hase and was developed at Daimler Chrysler for heavy duty diesel engines. The correlation works over a reasonably wide range of conditions and is given by

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$\tau_{id} = \left(\frac{C_1 + 0.22\bar{S}_p}{6N} \right) \exp\left[E_a \left(\frac{1}{RT} - \frac{1}{17,190} \right) + \left(\frac{21.2}{p - 12.4} \right)^{e_p} \right] \tag{10.11}$$

where τ_{id} is in seconds, C_1 is 0.36, N is engine speed in revolutions per minute, E_a is the effective activation energy and e_p is the pressure exponent. The expression for the effective activation energy is given by

$$E_a = \frac{E_{hh}}{CN + 25} \tag{10.12}$$

Table 10.1: Default Values of the Variables in the Hardenburg Correlation

Variable	E_{hh}	CN	C_1	e_p
Default	618,840	25	0.36	0.63

The second correlation, which is the generalized correlation, is given by Equation 10.9 (p. 425) and is available for ignition delay calculations.

10.2.3.3.3. Energy Release

If the ignition species is greater than or equal to 1 anywhere in the domain, ignition has occurred and combustion is no longer delayed. The ignition species acts as a switch to turn on the volumetric reactions in the domain. Note that the ignition species "mass fraction" can exceed 1 in the domain, therefore, it is not truly a mass fraction, but rather a passive scalar that represents the integrated correlation as a function of time.

10.3. Crevice Model

This section describes the theory behind the crevice model. Information can be found in the following sections:

10.3.1. Overview

10.3.2. Limitations

10.3.3. Crevice Model Theory

For information regarding using the crevice model, see Using the Crevice Model in the User's Guide.

10.3.1. Overview

The crevice model implemented in Ansys Fluent is a zero-dimensional ring-flow model based on the model outlined in Namazian and Heywood [467] (p. 1080) and Roberts and Matthews [555] (p. 1085). The model is geared toward in-cylinder specific flows, and more specifically, direct-injection (DI) diesel engines, and therefore is available only for time-dependent simulations.

The model takes mass, momentum, and energy from cells adjoining two boundaries and accounts for the storage of mass in the volumes of the crevices in the piston. Detailed geometric information regarding the ring and piston—typically a ring pack around the bore of an engine—is necessary to use the crevice model. An example representation is shown in Figure 10.3: Crevice Model Geometry (Piston) (p. 428) — Figure 10.5: Crevice Model "Network" Representation (p. 428).

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Figure 10.3: Crevice Model Geometry (Piston)

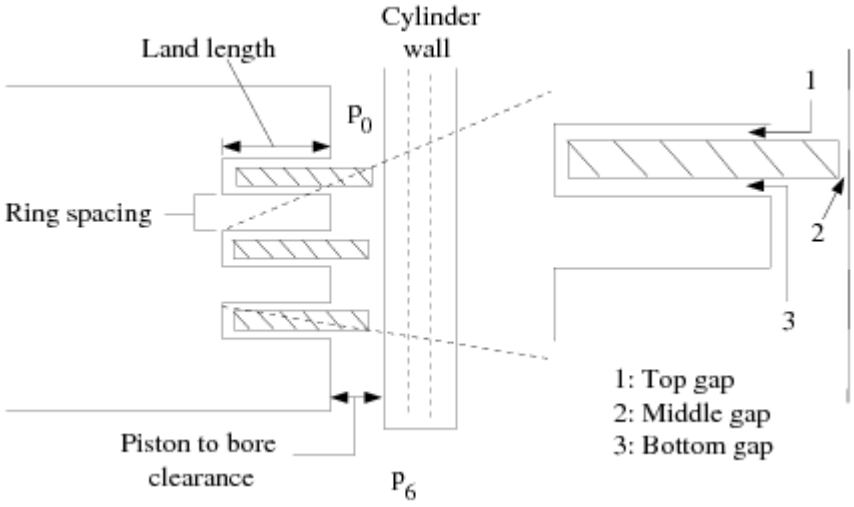


Figure 10.4: Crevice Model Geometry (Ring)

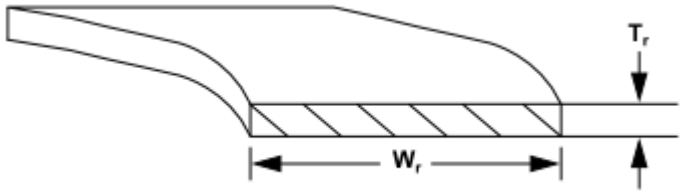
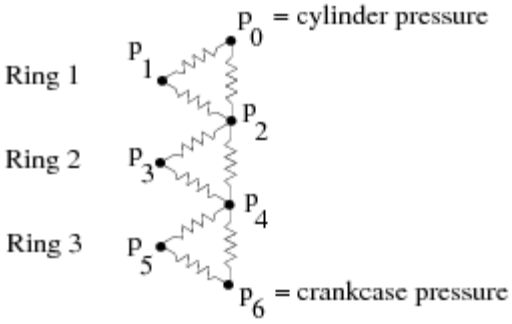


Figure 10.5: Crevice Model "Network" Representation



10.3.1.1. Model Parameters

- The *piston to bore clearance* is the distance between the piston and the bore. Typical values are 2 to 5 mil (80 to 120 μm) in a spark engine (SI) and 4 to 7 mil (100 to 240 μm) in some diesel engines (DI).
- The *ring thickness* is the variable T_r in Figure 10.4: Crevice Model Geometry (Ring) (p. 428). Typical values range from 1 to 3 mm for SI engines and 2 to 4 mm for DI engines.
- The *ring width* is the variable W_r in Figure 10.4: Crevice Model Geometry (Ring) (p. 428). Typical values range from 3 to 3.5 mm for SI engines and 4 to 6 mm for DI diesel engines.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

- The *ring spacing* is the distance between the bottom of one ring land and the top of the next ring land. Typical values of the ring spacing are 3 to 5 mm for SI engines and 4 to 8 mm for DI diesel engines.
- The *land length* is the depth of the ring land (that is, the cutout into the piston); always deeper than the width of the ring by about 1 mm. Typical values are 4 to 4.5 mm for SI engines and 5 to 7 mm for DI diesel engines.
- The *top gap* is the clearance between the ring land and the top of the ring (40 to 80 μm).
- The *middle gap* is the distance between the ring and the bore (10 to 40 μm).
- The *bottom gap* is the clearance between the ring land and the bottom of the ring (40 to 80 μm).
- The *shared boundary and leaking wall* is the piston (for example, `wall-8`) and the cylinder wall (for example, `wall.1`) in most in-cylinder simulations. Cells that share a boundary with the top of the piston and the cylinder wall are defined as the *crevice* cells.

The ring pack is the set of rings that seal the piston in the cylinder bore. As the piston moves upward in the cylinder when the valves are closed (for example, during the compression stroke in a four-stroke cycle engine), the pressure in the cylinder rises and flow begins to move past the rings. The pressure distribution in the ring pack is modeled by assuming either fully-developed compressible flow through the spaces between the rings and the piston, or choked compressible flow between the rings and the cylinder wall.

Since the temperature in the ring pack is fixed and the geometry is known, once a pressure distribution is calculated, the mass in each volume can be found using the ideal gas equation of state. The overall mass flow out of the ring pack (that is, the flow past the last ring specified) is also calculated at each discrete step in the Ansys Fluent solution.

10.3.2. Limitations

The limitations of the crevice model are that it is zero dimensional, transient, and currently limited to two threads that share a boundary.

A zero-dimensional approach is used because it is difficult to accurately predict lateral diffusion of species in the crevice. If the lateral diffusion of species is important in the simulation, as in when a spray plume in a DI engine is in close proximity to the boundary and the net mass flow is into the crevice, it is recommended that the full multidimensional crevice geometry be simulated in Ansys Fluent using a nonconformal mesh. Additionally, this approach does not specifically track individual species, as any individual species would be instantly distributed over the entire ring pack. The mass flux into the domain from the crevice is assumed to have the same composition as the cell into which mass is flowing.

The formulation of the crevice flow equations is inherently transient and is solved using Ansys Fluent's stiff-equation solver. A steady problem with leakage flow can be solved by running the transient problem to steady state. Additional limitations of the crevice model in its current form are that only a single crevice is allowed and only one thread can have leakage. Ring dynamics are not explicitly accounted for, although ring positions can be set during the simulation.

In this context, the crevice model solution is a stiff initial boundary-value problem. The stiffness increases as the pressure difference between the ring crevices increases and also as the overall pressure difference across the ring pack increases. Thus, if the initial conditions are very far from the solution

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

during a time step, the ODE solver may not be able to integrate the equations successfully. One solution to this problem is to decrease the flow time step for several iterations. Another solution is to start with initial conditions that are closer to the solution at the end of the time step.

10.3.3. Crevice Model Theory

Ansys Fluent solves the equations for mass conservation in the crevice geometry by assuming laminar compressible flow in the region between the piston and the top and bottom faces of the ring, and by assuming an orifice flow between the ring and the cylinder wall. The equation for the mass flow through the ring end gaps is of the form

$$\dot{m}_{ij} = C_d A_{ij} \rho c \eta_{ij} \tag{10.13}$$

where C_d is the discharge coefficient, A_{ij} is the gap area, ρ is the gas density, c is the local speed of sound, and η_{ij} is a compressibility factor given by

$$\eta_{ij} = \begin{cases} \frac{2}{\gamma - 1} \left[\left(\frac{p_i}{p_j} \right)^{\frac{2}{\gamma}} - \left(\frac{p_i}{p_j} \right)^{\frac{2}{\gamma}} \right]^{0.5} & \frac{p_i}{p_j} > 0.52 \\ \left(\frac{2}{\gamma - 1} \right)^{\frac{\gamma+1}{2(\gamma-1)}} & \frac{p_i}{p_j} \le 0.52 \end{cases} \tag{10.14}$$

where γ is the ratio of specific heats, p_i the upstream pressure and p_j the downstream pressure. The equation for the mass flow through the top and bottom faces of the ring (that is, into and out of the volume behind the piston ring) is given by

$$\dot{m}_{ij} = \frac{h_{ij}^2 \left(p_i^2 - p_j^2 \right) A_{ij}}{24 W_r \mu_{gas} R T} \tag{10.15}$$

where h_{ij} is the cross-sectional area of the gap, W_r is the width of the ring along which the gas is flowing, μ_{gas} is the local gas viscosity, T is the temperature of the gas and R is the universal gas constant. The system of equations for a set of three rings is of the following form:

$$\frac{dp_1}{dt} = \frac{p_1}{m_1} \left(\dot{m}_{01} - \dot{m}_{12} \right) \tag{10.16}$$

$$\frac{dp_2}{dt} = \frac{p_2}{m_2} \left(\dot{m}_{02} + \dot{m}_{12} - \dot{m}_{23} - \dot{m}_{24} \right) \tag{10.17}$$

$$\frac{dp_3}{dt} = \frac{p_3}{m_3} \left(\dot{m}_{23} - \dot{m}_{34} \right) \tag{10.18}$$

$$\frac{dp_4}{dt} = \frac{p_4}{m_4} \left(\dot{m}_{24} + \dot{m}_{34} - \dot{m}_{45} - \dot{m}_{46} \right) \tag{10.19}$$

$$\frac{dp_5}{dt} = \frac{p_5}{m_5} \left(\dot{m}_{45} - \dot{m}_{56} \right) \tag{10.20}$$

where p_0 is the average pressure in the crevice cells and p_6 is the crankcase pressure input from the text interface. The expressions for the mass flows for numerically adjacent zones (for example, 0-1, 1-2, 2-3, and so on) are given by Equation 10.15 (p. 430) and expressions for the mass flows for zones separated by two integers (for example, 0-2, 2-4, 4-6) are given by Equation 10.13 (p. 430) and Equation 10.14 (p. 430). Thus, there are $2n_r - 1$ equations needed for the solution to the ring-pack equations, where n_r is the number of rings in the simulation.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Chapter 11: Aerodynamically Generated Noise

The discipline of acoustics is intimately related to fluid dynamics. Many sounds that are technologically important in industrial applications are generated by and propagated in fluid flows. The phenomena associated with sounds can therefore be understood and analyzed in the general framework of fluid dynamics. (The governing equations for acoustics are indeed the same as the ones governing fluid flows.)

The main challenge in numerically predicting sound waves stems from the well-recognized fact that sounds have much lower energy than fluid flows, typically by several orders of magnitude. This poses a great challenge to the computation of sounds in terms of difficulty of numerically resolving sound waves, especially when one is interested in predicting sound propagation to the far field. Another challenge comes from the difficulty of predicting the very flow phenomena (for example, turbulence) in the near field that are responsible for generating sounds.

This chapter provides an overview and theoretical background of Ansys Fluent's approaches to computing aerodynamically generated sound. For more information about using the acoustics model, see Predicting Aerodynamically Generated Noise in the User's Guide .

11.1. Overview

11.2. Acoustics Model Theory

11.1. Overview

Considering the breadth of the discipline and the challenges encountered in aerodynamically generated noise, it is not surprising that a number of computational approaches have been proposed over the years whose sophistication, applicability, and cost widely vary.

Ansys Fluent offers three approaches to computing aerodynamically generated noise: a direct approach, a hybrid approach, and an approach that utilizes broadband noise source models. Within the hybrid approach Fluent offers the two methods, which are the integral method by Ffowcs Williams and Hawkings [168] (p. 1062) , and a differential sound propagation method based on a finite volume solver for a wave equation.

This section is divided into the following sections:

11.1.1. Direct Method

11.1.2. Integral Method by Ffowcs Williams and Hawkings

11.1.3. Method Based on Wave Equation

11.1.4. Broadband Noise Source Models

11.1.1. Direct Method

In this method, both generation and propagation of sound waves are directly computed by solving the appropriate fluid dynamics equations. Prediction of sound waves always requires time-accurate

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

solutions to the governing equations. Furthermore, in most practical applications of the direct method, one has to employ governing equations that are capable of modeling viscous and turbulence effects, such as unsteady Navier-Stokes equations (that is, DNS), RANS equations, and filtered equations used in SAS / DES / SDES / SBES and LES.

The direct method is therefore computationally difficult and expensive inasmuch as it requires highly accurate numerics, very fine computational meshes all the way to receivers, and acoustically non-reflecting boundary conditions. The computational cost becomes prohibitive when sound is to be predicted in the far field (for example, hundreds of chord-lengths in the case of an airfoil). The direct method becomes feasible when receivers are in the near field (for example, cabin noise). In many such situations involving near-field sound, sounds (or pseudo-sounds for that matter) are predominantly due to local hydrodynamic pressure, which can be predicted with a reasonable cost and accuracy.

Since sound propagation is directly resolved in this method, one normally needs to solve the compressible form of the governing equations (for example, compressible RANS equations, compressible form of filtered equations for LES). Only in situations where the flow is low and subsonic, and the receivers in the near field consist primarily of local hydrodynamic pressure fluctuations (that is, pseudo sound), can incompressible flow formulations be used. However, this incompressible treatment will not permit you to simulate resonance and feedback phenomena.

11.1.2. Integral Method by Ffowcs Williams and Hawkings

For predictions of mid- to far-field noise, the methods based on Lighthill's acoustic analogy [371] (p. 1074) offer viable alternatives to the direct method. In this approach, the near-field flow obtained from appropriate governing equations such as unsteady RANS equations, SAS / DES / SDES / SBES, or LES are used to predict the sound with the aid of analytically derived integral solutions to wave equations. The acoustic analogy essentially decouples the propagation of sound from its generation, enabling one to separate the flow solution process from the acoustics analysis.

Ansys Fluent offers a method based on the Ffowcs Williams and Hawkings (FW-H) formulation [178] (p. 1063) . The FW-H formulation adopts the most general form of Lighthill's acoustic analogy, and is capable of predicting sound generated by equivalent acoustic sources. Ansys Fluent adopts a time-domain integral formulation wherein time histories of sound pressure, or acoustic signals, at prescribed receiver locations are directly computed by evaluating corresponding surface integrals.

Time-accurate solutions of the flow-field variables, such as pressure, velocity components, and density on source (emission) surfaces, are required to evaluate the surface integrals. Time-accurate solutions can be obtained from unsteady Reynolds-averaged Navier-Stokes (URANS) equations, large eddy simulation (LES), or SAS / DES / SDES / SBES as appropriate for the flow at hand and the features that you want to capture (for example, vortex shedding). The source surfaces can be placed not only on impermeable walls, but also on interior (permeable) surfaces, which enables you to account for the contributions from the quadrupoles enclosed by the source surfaces. Both broadband and tonal noise can be predicted depending on the nature of the flow (noise source) being considered, turbulence model employed, and the time scale of the flow resolved in the flow calculation.

The FW-H acoustics model in Ansys Fluent allows you to select multiple source surfaces and receivers. Receivers can either be steady in the CFD reference frame or move with a user-specified constant velocity. The latter option enables modeling a "fly over" situation. The model also permits you either to save the source data for a future use, or to carry out an "on the fly" acoustic calculation simultaneously as the transient flow calculation proceeds, or both. Sound pressure signals thus obtained can be processed using the fast Fourier transform (FFT) and associated postprocessing capabilities to

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

compute and plot such acoustic quantities as the overall sound pressure level (SPL) and power spectra.

One important limitation of the integral FW-H formulation is that it is applicable only to predicting the propagation of sound toward *free space* . Thus, while the model can be legitimately used to predict far-field noise due to external aerodynamic flows, such as the flows around ground vehicles and aircraft, it cannot be used for predicting the noise propagation inside ducts or wall-enclosed space.

11.1.3. Method Based on Wave Equation

In this hybrid simulation method, which is intended to simulate aeroacoustics of low Mach number flows, the incompressible flow model is used for the calculation of sound sources, while the differential wave equation is used to calculate the propagation of sound generated by these sources. The acoustics wave equation implemented in Ansys Fluent is derived from the acoustics perturbation equations by Ewert and Schroeder [168] (p. 1062) under the assumption of constant density flow. The main advantages of this model are:

• Extended applicability compared to the Ffowcs Williams and Hawking integral solver, which can be used only to model the sound propagation in open space (see The Ffowcs Williams and Hawkings Model (p. 434)).

• Simple and convenient workflow of hybrid aeroacoustics simulations, without any data exchange between different software components via disk files, and without the interpolation of sound sources on a different mesh. A transient co-simulation of fluid flow and noise propagation using the same domain and mesh has been chosen for the current implementation.

The method includes the Kirchhoff integral tool to propagate the wave equation solution to the remote receivers in open space.

11.1.4. Broadband Noise Source Models

In many practical applications involving turbulent flows, noise does not have any distinct tones, and the sound energy is continuously distributed over a broad range of frequencies. In those situations involving *broadband noise* , statistical turbulence quantities readily computable from RANS equations can be utilized, in conjunction with semi-empirical correlations and Lighthill's acoustic analogy, to shed some light on the source of broadband noise.

Ansys Fluent offers several such *source models* that enable you to quantify the local contribution (per unit surface area or volume) to the total acoustic power generated by the flow. They include the following:

- Proudman's formula
- jet noise source model
- boundary layer noise source model
- source terms in the linearized Euler equations
- source terms in Lilley's equation

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Considering that one would ultimately want to come up with some measures to mitigate the noise generated by the flow in question, the source models can be employed to extract useful diagnostics on the noise source to determine which portion of the flow is primarily responsible for the noise generation. Note, however, that these source models do not predict the sound at receivers.

Unlike the direct method and the FW-H integral method, the broadband noise source models do not require transient solutions to any governing fluid dynamics equations. All source models require what typical RANS models would provide, such as the mean velocity field, turbulent kinetic energy (k) and the dissipation rate (ε). Therefore, the use of broadband noise source models requires the least computational resources.

11.2. Acoustics Model Theory

This section describes the theoretical background for the Ffowcs Williams and Hawkings model, the wave equation model, and the broadband noise source models.

This section is divided into the following sections:

11.2.1. The Ffowcs Williams and Hawkings Model

11.2.2. Wave Equation Model

11.2.3. Broadband Noise Source Models

11.2.1. The Ffowcs Williams and Hawkings Model

The Ffowcs Williams and Hawkings (FW-H) equation is essentially an inhomogeneous wave equation that can be derived by manipulating the continuity equation and the Navier-Stokes equations. The FW-H [79] (p. 1057) , [178] (p. 1063) equation can be written as:

$$\begin{aligned} \frac{1}{a_0^2}\frac{\partial^2 p'}{\partial t^2} - \nabla^2 p' &= \frac{\partial^2}{\partial x_i \partial x_j}\{T_{ij}H(f)\} \\ &- \frac{\partial}{\partial x_i}\{[P_{ij}n_j + \rho u_i(u_n - v_n)]\,\delta(f)\} \\ &+ \frac{\partial}{\partial t}\{[\rho_0 v_n + \rho(u_n - v_n)]\,\delta(f)\} \end{aligned} \tag{11.1}$$

where

u_i = fluid velocity component in the x_i direction

u_n = fluid velocity component normal to the surface $f=0$

v_i = surface velocity components in the x_i direction

v_n = surface velocity component normal to the surface

$\delta(f)$ = Dirac delta function

$H(f)$ = Heaviside function

p' is the sound pressure at the far field ($p'=p-p_0$). $f=0$ denotes a mathematical surface introduced to "embed" the exterior flow problem ($f>0$) in an unbounded space, which facilitates the use of generalized function theory and the free-space Green function to obtain the solution. The surface ($f=0$) corresponds to the source (emission) surface, and can be made coincident with a body (impermeable) surface or a permeable surface off the body surface. n_i is the unit normal vector pointing

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

toward the exterior region ($f > 0$), a_0 is the far-field sound speed, and T_{ij} is the Lighthill stress tensor, defined as

$$T_{ij}=\rho u_i u_j+P_{ij}-a_0^2\left(\rho-\rho_0\right)\delta_{ij} \tag{11.2}$$

P_{ij} is the compressive stress tensor. For a Stokesian fluid, this is given by

$$P_{ij}=p\delta_{ij}-\mu\left[\frac{\partial u_i}{\partial x_j}+\frac{\partial u_j}{\partial x_i}-\frac{2}{3}\frac{\partial u_k}{\partial x_k}\delta_{ij}\right] \tag{11.3}$$

The free-stream quantities are denoted by the subscript 0 .

The wave equation Equation 11.1 (p. 434) can be integrated analytically under the assumptions of the free-space flow and the absence of obstacles between the sound sources and the receivers. The complete solution consists of surface integrals and volume integrals. The surface integrals represent the contributions from monopole and dipole acoustic sources and partially from quadrupole sources, whereas the volume integrals represent quadrupole (volume) sources in the region outside the source surface. The contribution of the volume integrals becomes small when the flow is low subsonic and the source surface encloses the source region. In Ansys Fluent, the volume integrals are dropped. Thus, we have

$$p'\left(\vec{x},t\right)=p'_T\left(\vec{x},t\right)+p'_L\left(\vec{x},t\right) \tag{11.4}$$

where

$$\begin{aligned} 4\pi p'_T\left(\vec{x},t\right) &= \int_{f=0}\left[\frac{\rho_0\left(\dot{U}_n+U_{\dot{n}}\right)}{r\left(1-M_r\right)^2}\right]dS \\ &+ \int_{f=0}\left[\frac{\rho_0 U_n\left\{r\dot{M}_r+a_0\left(M_r-M^2\right)\right\}}{r^2\left(1-M_r\right)^3}\right]d \end{aligned} \tag{11.5}$$

$$\begin{aligned} 4\pi p'_L\left(\vec{x},t\right) &= \frac{1}{a_0}\int_{f=0}\left[\frac{\dot{L}_r}{r\left(1-M_r\right)^2}\right]dS \\ &+ \int_{f=0}\left[\frac{L_r-L_M}{r^2\left(1-M_r\right)^2}\right]dS \\ &+ \frac{1}{a_0}\int_{f=0}\left[\frac{L_r\left\{r\dot{M}_r+a_0\left(M_r-M^2\right)\right\}}{r^2\left(1-M_r\right)^3}\right]dS \end{aligned} \tag{11.6}$$

where

$$U_i = v_i+\frac{\rho}{\rho_0}\left(u_i-v_i\right) \tag{11.7}$$

$$L_i = P_{ij}\hat{n}_j+\rho u_i\left(u_n-v_n\right) \tag{11.8}$$

When the integration surface coincides with an impenetrable wall, the two terms on the right in Equation 11.4 (p. 435) , $p'_T\left(\vec{x},t\right)$ and $p'_L\left(\vec{x},t\right)$, are often referred to as thickness and loading terms, respectively, in light of their physical meanings. The square brackets in Equation 11.5 (p. 435) and Equation 11.6 (p. 435) denote that the kernels of the integrals are computed at the corresponding retarded times, τ , defined as follows, given the receiver time, t , and the distance to the receiver, r ,

$$\tau=t-\frac{r}{a_0} \tag{11.9}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

The various subscripted quantities appearing in Equation 11.5 (p. 435) and Equation 11.6 (p. 435) are the inner products of a vector and a unit vector implied by the subscript. For instance, $L_r=\vec{L}\cdot\hat{\vec{r}}=L_i r_i$ and $U_n=\vec{U}\cdot\vec{n}=U_i n_i$, where $\vec{r}$ and $\vec{n}$ denote the unit vectors in the radiation and wall-normal directions, respectively. The Mach number vector M_i in Equation 11.5 (p. 435) and Equation 11.6 (p. 435) relates to the motion of the integration surface: $M_i=v_i/a_o$. The L_M quantity is a scalar product $L_i M_i$. The dot over a variable denotes source-time differentiation of that variable.

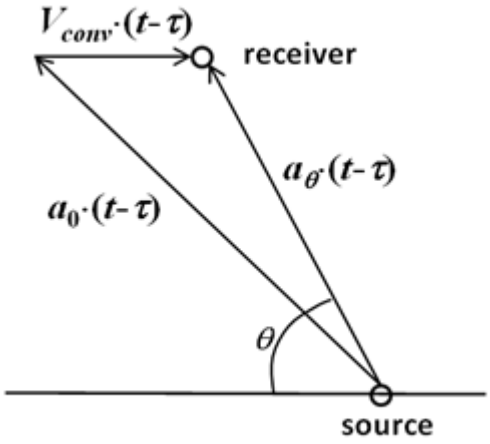
For the calculation of the aerodynamic sound caused by external flow around a body, the **Convective Effects** option must be enabled (see Setting Model Constants in the User's Guide) with the far-field fluid velocity vector $\vec{V}_{conv}$ to be additionally specified. This option is relevant to practical situations such as the flight tests (microphones mounted on an airplane) or wind tunnel measurements (microphones mounted within the wind tunnel core flow region).

With the convective effects taken into account, the retarded time calculation becomes more complicated than the simple form of Equation 11.9 (p. 435) . As illustrated in Figure 11.1: Schematic of the Convective Effect on the Retarded Time Calculation (p. 436) , the effective sound propagation velocity a_θ is lower than a_0 in the upstream hemisphere and higher than a_0 in the downstream hemisphere, where θ is the direction angle towards the receiver counted from the upstream direction. According to Figure 11.1: Schematic of the Convective Effect on the Retarded Time Calculation (p. 436) the retarded time with the convective effect must be computed as

$$\tau=t-r/a_\theta\,,\quad a_\theta=\sqrt{a_0^2-V_{conv}^2\sin^2\theta}-V_{conv}\cos\theta \tag{11.10}$$

An obvious limitation for the **Convective Effects** option is that the far-field convection must be subsonic $(V_{conv}<a_0)$

Figure 11.1: Schematic of the Convective Effect on the Retarded Time Calculation



Note the following remarks regarding the applicability of the FW-H integral solution:

- The FW-H formulation in Ansys Fluent can handle rotating surfaces as well as stationary surfaces.
- It is not required that the surface $f=0$ coincide with body surfaces or walls. The formulation permits source surfaces to be permeable, and therefore can be placed in the interior of the flow.
- When a permeable source surface (either interior or nonconformal sliding interface) is placed at a certain distance off the body surface, the integral solutions given by Equation 11.5 (p. 435) and

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Equation 11.6 (p. 435) include the contributions from the quadrupole sources within the region enclosed by the source surface. When using a permeable source surface, the mesh resolution must be fine enough to resolve the transient flow structures inside the volume enclosed by the permeable surface.

11.2.2. Wave Equation Model

Like the Ffowcs Williams and Hawkings (FW-H) integral model, this model based on the finite volume solver of the wave equation, obtains information about the sound sources from the calculation of a background flow, which generates sound. Different from the FW-H model, which calculates sound signals at selected receiver locations, the wave equation model can deliver a sound field in the entire simulation domain. In this sense, this model is similar to the direct simulation approach (see Direct Method (p. 431)). However, unlike the direct simulation, the wave equation model allows you to see a distribution of the sound pressure separately from the flow pressure.

11.2.2.1. Limitations

The following limitations apply:

- The background flow must be a single phase at low Mach number without significant variations of density or speed of sound.
- The effect of convection is neglected by the wave equation.
- The 2-D axisymmetric version of Fluent is not supported.
- The transient scheme must be the implicit scheme with either the second order or bounded second order enabled (the explicit scheme and the first order implicit time scheme for the flow solver are not supported).
- The domain and mesh must be stationary. Mesh and frame motion is not supported.
- Overset meshes are not supported.
- The User Defined Function of the type `DEFINE_INIT` `DEFINE_INIT` cannot be used to initialize the wave equation model. Please use instead the `DEFINE_ON_DEMAND` function `DEFINE_ON_DEMAND` .

11.2.2.2. Governing Equations and Boundary Conditions

Written for the sound potential, φ , the wave equation has the following form:

$$\frac{\partial^2 \varphi}{\partial t^2} - c^2 \nabla^2 \varphi = \frac{1}{\rho} \frac{\partial}{\partial t} p_{flow} \tag{11.11}$$

Where p_{flow} is the local instant value of the flow static pressure, ρ is the constant fluid density, and c is the constant speed of sound.

At low Mach numbers, the convective transport of sound may be neglected, so the sound pressure p' is computed from the sound potential using a simplified relation:

$$p' = -\rho \frac{\partial \varphi}{\partial t} \tag{11.12}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

The following boundary condition types for the wave equation are currently available in Fluent:

- Walls: all walls are considered acoustically hard, ideally reflecting sound.
- Permeable boundaries (inlets, outlets, far-field boundaries): these boundaries are handled using non-reflective conditions.
- Symmetry and periodic boundaries.

Impedance boundary conditions are not yet available for the wave equation model.

Non-conformal stationary mesh interfaces can be used with the acoustics models. However, local wall zones, which may be created on such interfaces to handle significant geometrical misalignments, reflect sound and may strongly influence the acoustics solution. Therefore it is recommended to avoid such non-physical walls, unless they are covered by a sponge region.

11.2.2.3. Method of Numerical Solution

The Laplacian term in Equation 11.11 (p. 437) is discretized in space using the same finite volume procedure that is applied in Fluent for the diffusion terms of the flow equations.

The solution is propagated in time using the α -method by Hilber, Hughes, and Taylor, which is also used for transient analysis by Mechanical APDL and is documented in the Ansys Mechanical APDL Theory Reference, section 15.2.2.1: Time Integration Scheme for Linear Systems. This method is an advanced variant of the popular algorithm by Newmark, which is used for numerical integration of evolutionary equations with the leading transient term represented by a second order time derivative, like the wave equation in Equation 11.11 (p. 437) . It combines second order accuracy with good stability.

11.2.2.4. Preventing Non-Physical Reflections of Sound Waves

The following sections detail some strategies and considerations to prevent non-physical reflections of sound waves, which have a large influence on solution accuracy.

11.2.2.4.1. Mesh Quality

The numerical solution of the wave equation is known to be very sensitive to any non-uniformities in the mesh that have a size of the order of the sound wave length or larger. Preventing such non-uniformities is commonly recommended and it is easier to estimate the size of a dangerous non-uniformity in the mesh when performing a harmonic analysis of acoustics in the frequency domain. For the time-domain aeroacoustics simulation, estimation of the mesh quality criteria is not as easy because of the broad-band nature of the vortex sound in a turbulent flow. A scale-resolving simulation normally resolves fluctuations down to the mesh cell size, and it may not be obvious how to estimate the dangerous size of mesh non-uniformities. Results of aeroacoustics simulations in the time domain are often less accurate in the high-frequency part of spectrum due to the reflection and dispersion of short waves at such non-uniformities. These inaccuracies can be reduced by the use of the numerical procedures described below in this section. Still, the mesh quality requirements for aeroacoustics simulations are higher than those for incompressible flow simulations in similar configurations. It is recommended to estimate in advance the frequency range of interest, and allocate 12 or more cells per wavelength within this range.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

11.2.2.4.2. Filtering of the Sound Source Term

Two kinds of filters are applied to smooth the source term of Equation 11.11 (p. 437) in space and time: a second order explicit linear filter in space suppresses odd-even spatial oscillations, and a high-order explicit filter in time damps the high frequency part of the source spectrum. The space filter is always active, and can be switched off only by changing an rp-variable. As for the time filter, there is a graphical and text user interface to activate or de-activate it. It is recommended to keep the time filter active, which is the default setting.

11.2.2.4.3. Ramping in Time and Limiting in Space (Masking) of the Sound Source Term

The acoustics wave equation model should be activated after the time-dependent flow solution has been well established. This means a flow simulation needs to be stopped and then continued with the wave equation solver switched on. However, a sudden activation of the sound source term in the wave equation can create non-physical solution disturbances of high magnitude, which may remain in the domain for a very long time. To avoid this, a slow ramping of the source is applied in the wave equation solver, with the duration specified by the user in terms of number of timesteps. A common rule to estimate this duration is to make it equal to one or few periods of the low frequency sound which is significant and intended to be well-resolved. During the ramping period, the source term is pre-multiplied by a factor, which grows smoothly in time from zero to one.

Another measure to improve the solution quality is limiting (masking) the sound source region. It serves two purposes:

- allows only the desired part of the turbulent flow region to generate sound, where the solution is known to be well resolved.
- smoothly reduce the sound sources at the borders of the masking region to further improve the solution quality. This is achieved by a smooth transition of the masking marker between zero and one through the user-specified transition thickness. A value of zero means that the model source term is locally active, and the value of one means that the source is completely masked out.

11.2.2.4.4. Damping of Solution in a Sponge Region Using Artificial Viscosity

Application of a sponge region helps to avoid any non-physical sound reflection in the following situations:

- At open boundaries, there may still be some limited reflection despite the use of a non-reflective boundary condition, if the sound wave propagation direction is not normal to the boundary surface.
- In large domains, mesh cell size expansion in the directions away from sound sources is often necessary. Reflections from expanding mesh layers, and especially from internal mesh interfaces, may negatively affect the acoustics solution quality. It is recommended you avoid such mesh expansions and interfaces, which create overly coarse meshes within the acoustics region of interest for important sound frequencies. Outside this region, mesh irregularities should be covered by a sponge region. Similar to the source masking region, the border of the sponge region is smeared with a user-specified transition thickness.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

The damping of the wave equation solution is realized by adding an artificial viscous term to Equation 11.11 (p. 437) , which has the same form as a physical viscous term. The artificial viscosity value is selected depending on the local mesh resolution. Inside the user-specified sponge region, a viscous term with the default non-dimensional artificial viscosity coefficient, $\nu=1$, is optimized to most efficiently damp the shortest resolved waves with the wavelength equal to two cell sizes. If there is a need for stronger damping of longer waves, you can increase the artificial viscosity using the two parameters, C and δ :

$$\nu(\mathbf{r})=C\cdot f(\mathbf{r})+\delta \tag{11.13}$$

where $f(\mathbf{r})$ is the location-dependent sponge layer marker, which varies between zero and one.

The parameter δ is a base level for the artificial viscosity, which is used to smooth the solution everywhere in the acoustics region, meaning regions outside of the sponge region are also damped by this base level. The default values for the damping parameters are C = 1 and δ = 0.1.

11.2.2.5. Kirchhoff Integral

The acoustics wave equation model in Fluent is mainly intended to simulate sound propagation in the mid-field range. This range covers sound sources and the non-uniform flow field, where also sound-reflecting solid obstacles may exist. In many practical applications, there is a need to calculate the perceived sound signals at remote target locations much further away. Propagation of sound through open space, where fluid is either stagnant or moves uniformly, can be calculated using the Kirchhoff surface integral.

11.2.2.5.1. Compatibility and Limitations

The Kirchhoff integral model is a complementary model to the wave equation model and can be applied whenever the wave equation model is used. The current wave equation model can be applied to transient single-phase flow simulations, when the background flow does not cause density changes (constant density fluid model is recommended).

Limitations of the Kirchhoff model, which correspond to the limitations of the wave equation model in Fluent, are:

- Effect of convection is not included, meaning that the far field fluid must be stagnant.
- Motion of the integration surface (in particular its rotation) is not supported.

Additional limitations of the current implementation of the Kirchhoff model are:

- Only a 3-D variant is implemented.
- Only a single surface can be used for the integration. A complex surface can be built as an iso-surface of a correspondingly defined scalar field.
- Periodicity is not supported.
- Moving receivers are not supported.
- Model is available only for the “on-the-fly” mode, which means that every next set of signal samples is computed during a transient simulation after each time step. Currently there is no

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

"read-and-compute" mode such as for the Ffowcs Williams and Hawkings model, where the transient data are exported during the simulation, and Fluent performs a single post-processing action after the simulation end to compute the sound propagation to remote receivers.

11.2.2.5.2. Mathematical Formulation

Due to the linearity of the wave equation model and its integral solution by Kirchhoff, the surface integral may be written for any acoustics field variable. Since the currently implemented wave equation model uses the acoustic potential as the primary variable, the Kirchhoff integral is directly applied to potential:

$$\varphi(\mathbf{x},t)=\frac{1}{4\pi}\iint\left[(\mathbf{e}_R\cdot\mathbf{n})\left(\frac{\partial\varphi/\partial\tau}{cR}+\frac{\varphi}{R^2}\right)-\frac{\partial\varphi/\partial\mathrm{n}}{R}\right]dS \tag{11.14}$$

Where $\mathbf{x}$ and t are the receiver coordinates and the reception time, τ is the emission (retarded) time, R is the distance from a location on the integration surface to the receiver, $\mathbf{e}_R$ is the unit vector pointing to the receiver, $\mathbf{n}$ is the outward surface unit normal vector, and c is the speed of sound. Each elementary facet of the integration surface contributes to the integral at its own retarded time $\tau=t-R/c$. Integration is performed in Equation 11.14 (p. 441) over a user-specified surface, which theoretically must be closed. In practice, however, non-closed integration surfaces are applied in situations, where the Kirchhoff sound source (a function under the integral in Equation 11.14 (p. 441)) is either not known or may be neglected on a portion of a surface. One example is a long cylindrical surface without the end faces, which is used to compute sound, generated by a long cylinder in a cross flow. Another example is a spherical surface placed around the outlet of a duct or pipe in order to calculate sound, radiated from the orifice. In the latter case a part of a sphere, which is covered by a duct or pipe, is omitted in the integral.

After the acoustic potential samples are accumulated for the intended duration at all receivers, the sound pressure signal $p(t)$ is calculated at each receiver as:

$$p(t)=-\rho\frac{\partial\varphi}{\partial t} \tag{11.15}$$

11.2.3. Broadband Noise Source Models

11.2.3.1. Proudman's Formula

Proudman [533] (p. 1084) , using Lighthill's acoustic analogy, derived a formula for acoustic power generated by isotropic turbulence without mean flow. More recently, Lilley [372] (p. 1074) re-derived the formula by accounting for the retarded time difference that was neglected in Proudman's original derivation. Both derivations yield acoustic power due to the unit volume of isotropic turbulence (in $\mathrm{W/m^3}$) as

$$P_A=\alpha\rho_0\left(\frac{u^3}{\ell}\right)\frac{u^5}{a_0^5} \tag{11.16}$$

where u and ℓ are the turbulence velocity and length scales, respectively, and a_0 is the speed of sound. α in Equation 11.16 (p. 441) is a model constant. In terms of k and ε , Equation 11.16 (p. 441) can be rewritten as

$$P_A=\alpha_\varepsilon\rho_0\varepsilon M_t^5 \tag{11.17}$$

where

$$M_t = \frac{\sqrt{2k}}{a_0} \tag{11.18}$$

The rescaled constant, α_ε , is set to 0.1 in Ansys Fluent based on the calibration of Sarkar and Hussaini [564] (p. 1085) in using direct numerical simulation of isotropic turbulence.

Ansys Fluent can also report the acoustic power in dB, which is computed from

$$L_P = 10\log\left(\frac{P_A}{P_{ref}}\right) \tag{11.19}$$

where P_{ref} is the reference acoustic power (P_{ref} =10 $^{-12}$ W/m ^{3} by default).

The Proudman's formula gives an approximate measure of the local contribution to total acoustic power per unit volume in a given turbulence field. Proper caution, however, should be taken when interpreting the results in view of the assumptions made in the derivation, such as high Reynolds number, small Mach number, isotropy of turbulence, and zero mean motion.

11.2.3.2. The Jet Noise Source Model

This source model for axisymmetric jets is based on the works of Goldstein [210] (p. 1065) who modified the model originally proposed by Ribner [552] (p. 1085) to better account for anisotropy of turbulence in axisymmetric turbulent jets.

In Goldstein's model, the total acoustic power emitted by the unit volume of a turbulent jet is computed from

$$P_A(\vec{y}) = \int_{2\pi}^{0}\int_{\pi}^{0} I(r,\theta;\vec{y}) r^2 \sin\theta d\theta d\psi = 2\pi r^2 \int_{\pi}^{0} I(r,\theta;\vec{y}) \sin\theta d\theta \tag{11.20}$$

where r and θ are the radial and angular coordinates of the receiver location, and $I(r,\theta;\vec{y})$ is the directional acoustic intensity per unit volume of a jet defined by

$$I(r,\theta;\vec{y}) = \frac{12\rho_0\omega_f^4 L_1 L_2^2 \overline{u_{t1}^2}^2}{5\pi a_0^5 r^2}\frac{D_{self}}{C^5} + \frac{24\rho_0\omega_f^4 L_1 L_2^4 \overline{u_{t1}^2}}{\pi a_0^5 r^2}\left(\frac{\partial U}{\partial r}\right)^2 \frac{D_{shear}}{C^5} \tag{11.21}$$

C in Equation 11.21 (p. 442) is the modified convection factor defined by

$$C = 1 - M_c\cos\theta \tag{11.22}$$

and

$$D_{self} = 1 + 2\left(\frac{M}{9} - N\right)\cos^2\theta\sin^2\theta + \frac{1}{3}\left[\frac{M^2}{7} + M - 1.5N\left(3 - 3N + \frac{1.5}{\Delta^2} - \frac{\Delta^2}{2}\right)\right]\sin^4\theta \tag{11.23}$$

$$D_{shear} = \cos^2\theta\left[\cos^2\theta + \frac{1}{2}\left(\frac{1}{\Delta^2} - 2N\right)\sin^2\theta\right] \tag{11.24}$$

The remaining parameters are defined as

$$\Delta = \frac{L_2}{L_1} \tag{11.25}$$

$$M = \left[\frac{3}{2}\left(\Delta - \frac{1}{\Delta}\right)\right]^2 \tag{11.26}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$N=1-\frac{\left(\overline{u_{t2}^2}\right)}{\left(\overline{u_{t1}^2}\right)} \tag{11.27}$$

$$L_1=\frac{\left(\overline{u_{t1}^2}\right)^{3/2}}{\epsilon} \tag{11.28}$$

$$L_2=\frac{\left(\overline{u_{t2}^2}\right)^{3/2}}{\epsilon} \tag{11.29}$$

$$\omega_f=2\pi\frac{\epsilon}{k} \tag{11.30}$$

where $\overline{u_{t1}^2}$ and $\overline{u_{t2}^2}$ are computed differently depending on the turbulence model chosen for the computation. When the RSM is selected, they are computed from the corresponding normal stresses. For all other two-equation turbulence models, they are obtained from

$$\overline{u_{t1}^2} = \frac{8}{9}k \tag{11.31}$$

$$\overline{u_{t2}^2} = \frac{4}{9}k \tag{11.32}$$

Ansys Fluent reports the acoustic power both in the dimensional units (W/m ^{3}) and in dB computed from

$$L_P=10\log\left(\frac{P_A}{P_{ref}}\right) \tag{11.33}$$

where P_{ref} is the reference acoustic power (P_{ref} =10 $^{-12}$ W/m ^{3} by default).

11.2.3.3. The Boundary Layer Noise Source Model

Far-field sound generated by turbulent boundary layer flow over a solid body at low Mach numbers is often of practical interest. The Curle's integral [128] (p. 1060) based on acoustic analogy can be used to approximate the local contribution from the body surface to the total acoustic power. To that end, one can start with the Curle's integral

$$p'(\vec{x},t)=\frac{1}{4\pi a_0}\int_S\frac{\left(x_i-y_i\right)n_i}{r^2}\frac{\partial p}{\partial t}(\vec{y},\tau)dS(\vec{y}) \tag{11.34}$$

where τ denotes the emission time ($\tau=t-r/\ a_0$), and S the integration surface.

Using this, the sound intensity in the far field can then be approximated by

$$\overline{p'^2}\approx\frac{1}{16\pi^2a_0^2}\int_S\frac{\cos^2\theta}{r^2}\overline{\left[\frac{\partial p}{\partial t}(\vec{y},\tau)\right]^2}\mathcal{A}_c(\vec{y})dS(\vec{y}) \tag{11.35}$$

where $\mathcal{A}_c$ is the correlation area, $r\equiv\left|\vec{x}-\vec{y}\right|$, and θ is the angle between $\left|\vec{x}-\vec{y}\right|$ and the wall-normal direction $\vec{n}$.

The total acoustic power emitted from the entire body surface can be computed from

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$\begin{aligned} P_A &= \frac{1}{\rho_0 a_0}\int_0^{2\pi}\int_0^{\pi}\overline{p'^2}r^2\sin\theta d\theta d\psi \\ &= \int_S I(\vec{y})dS(\vec{y}) \end{aligned} \tag{11.36}$$

where

$$I(\vec{y}) \equiv \frac{\mathcal{A}_c(\vec{y})}{12\rho_0\pi a_0^3}\overline{\left[\frac{\partial p}{\partial t}\right]^2} \tag{11.37}$$

which can be interpreted as the local contribution per unit surface area of the body surface to the total acoustic power. The mean-square time derivative of the surface pressure and the correlation area are further approximated in terms of turbulent quantities like turbulent kinetic energy, dissipation rate, and wall shear.

Ansys Fluent reports the acoustic surface power defined by Equation 11.37 (p. 444) both in physical (W/m^2) and dB units.

11.2.3.4. Source Terms in the Linearized Euler Equations

The linearized Euler equations (LEE) can be derived from the Navier-Stokes equations starting from decompositions of the flow variables into mean, turbulent, and acoustic components, and by assuming that the acoustic components are much smaller than the mean and turbulent components. The resulting linearized Euler equations for the acoustic velocity components can be written as

$$\begin{aligned} \frac{\partial u_{ai}}{\partial t} + U_j\frac{\partial u_{ai}}{\partial x_j} + u_{aj}\frac{\partial U_i}{\partial x_j} + \frac{1}{\bar{\rho}}\frac{\partial p_a}{\partial x_i} - \frac{\rho_a}{\bar{\rho}^2}\frac{\partial P}{\partial x_i} = & \underbrace{-U_j\frac{\partial u_i'}{\partial x_j} - u_j'\frac{\partial U_i}{\partial x_j}}_{L_{sh}} \underbrace{-u_j'\frac{\partial u_i'}{\partial x_j}}_{L_{se}} \\ & -\frac{1}{\bar{\rho}}\frac{\partial p'}{\partial x_i} - \frac{\partial u_i'}{\partial t} + \frac{\partial}{\partial x_j}\overline{u_j'u_i'} \end{aligned} \tag{11.38}$$

where the subscript " a " refers to the corresponding acoustic components, and the prime superscript refers to the turbulent components.

The right side of Equation 11.38 (p. 444) can be considered as effective source terms responsible for sound generation. Among them, the first three terms involving turbulence are the main contributors. The first two terms denoted by L_{sh} are often referred to as "shear-noise" source terms, since they involve the mean shear. The third term denoted by L_{se} is often called the "self-noise" source term, as it involves turbulent velocity components only.

The turbulent velocity field needed to compute the LEE source terms is obtained using the method of stochastic noise generation and radiation (SNGR) [55] (p. 1056) . In this method, the turbulent velocity field and its derivatives are computed from a sum of N Fourier modes.

$$\vec{u}(\vec{x},t) = 2\sum_{n=1}^{N}\tilde{u}_n\cos\left(\vec{k}_n\cdot\vec{x}+\psi_n\right)\vec{\sigma}_n \tag{11.39}$$

where $\tilde{u}_n$, ψ_n , $\vec{\sigma}_n$ are the amplitude, phase, and directional (unit) vector of the n^{th} Fourier mode associated with the wave-number vector $\vec{k}_n$.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Note that the source terms in the LEE are vector quantities, having two or three components depending on the dimension of the problem at hand.

11.2.3.5. Source Terms in Lilley's Equation

Lilley's equation is a third-order wave equation that can be derived by combining the conservation of mass and momentum of compressible fluids. When the viscous terms are omitted, it can be written in the following form:

$$\frac{D}{Dt}\left[\frac{D^2\Pi}{Dt^2}-\frac{\partial}{\partial x_j}\left(a^2\frac{\partial\Pi}{\partial x_j}\right)\right]+2\frac{\partial u_k}{\partial x_j}\frac{\partial}{\partial x_k}\left(a^2\frac{\partial\Pi}{\partial x_j}\right)=-2\frac{\partial u_k}{\partial x_i}\frac{\partial u_j}{\partial x_k}\frac{\partial u_i}{\partial x_j} \tag{11.40}$$

where $\Pi=\left(1/\gamma\right)\ln\frac{p}{p_o}$.

Lilley's equation can be linearized about the underlying steady flow as

$$u_i\left(\vec{x},t\right)=U_i\left(\vec{x}\right)+u_i'\left(\vec{x},t\right) \tag{11.41}$$

where $u'\left(\vec{x},t\right)$ is the turbulent velocity component.

Substituting Equation 11.41 (p. 445) into the source term of Equation 11.40 (p. 445) , we have

$$\begin{aligned} S &\equiv -2\frac{\partial u_k}{\partial x_i}\frac{\partial u_j}{\partial x_k}\frac{\partial u_i}{\partial x_j} \\ &= -2\frac{\partial U_k}{\partial x_i}\frac{\partial U_j}{\partial x_k}\frac{\partial U_i}{\partial x_j}\underbrace{-2\frac{\partial u_k'}{\partial x_i}\frac{\partial u_j'}{\partial x_k}\frac{\partial u_i'}{\partial x_j}}_{\text{Self-Noise Terms}}\underbrace{-6\frac{\partial U_k}{\partial x_i}\frac{\partial U_j}{\partial x_k}\frac{\partial u_i'}{\partial x_j}-6\frac{\partial u_k'}{\partial x_i}\frac{\partial u_j'}{\partial x_k}\frac{\partial U_i}{\partial x_j}}_{\text{Shear-Noise Terms}} \end{aligned} \tag{11.42}$$

The resulting source terms in Equation 11.42 (p. 445) are evaluated using the mean velocity field and the turbulent (fluctuating) velocity components synthesized by the SNGR method. As with the LEE source terms, the source terms in Equation 11.42 (p. 445) are grouped depending on whether the mean velocity gradients are involved (*shear noise* or *self noise*), and reported separately in Ansys Fluent.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information
of ANSYS, Inc. and its subsidiaries and affiliates.

Chapter 12: Discrete Phase

This chapter describes the theory behind the Lagrangian discrete phase capabilities available in Ansys Fluent. For information about how to use discrete phase models, see Modeling Discrete Phase in the User's Guide. Information in this chapter is organized into the following sections:

12.1. Introduction

12.2. Particle Motion Theory

12.3. Laws for Drag Coefficients

12.4. Laws for Heat and Mass Exchange

12.5. Vapor Liquid Equilibrium Theory

12.6. Physical Property Averaging

12.7. Wall-Particle Reflection Model Theory

12.8. Wall-Jet Model Theory

12.9. Lagrangian Wall-Film Model Theory

12.10. Wall Erosion

12.11. Particle–Wall Impingement Heat Transfer Theory

12.12. Atomizer Model Theory

12.13. Secondary Breakup Model Theory

12.14. Collision and Droplet Coalescence Model Theory

12.15. Discrete Element Method Collision Model

12.16. One-Way and Two-Way Coupling

12.17. Node Based Averaging

12.18. Blockage Effect

12.1. Introduction

Advances in computational fluid mechanics have provided the basis for further insight into the dynamics of multiphase flows. Currently there are two approaches for the numerical calculation of multiphase flows: the Euler-Lagrange approach (discussed below) and the Euler-Euler approach (discussed in Approaches to Multiphase Modeling (p. 598)).

For more information, see the following section:

12.1.1. The Euler-Lagrange Approach

12.1.1. The Euler-Lagrange Approach

The Lagrangian discrete phase model in Ansys Fluent (described in this chapter) follows the Euler-Lagrange approach. The fluid phase is treated as a continuum by solving the Navier-Stokes equations,

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

while the dispersed phase is solved by tracking a large number of particles, bubbles, or droplets through the calculated flow field. The dispersed phase can exchange momentum, mass, and energy with the fluid phase.

This approach is made considerably simpler when particle-particle interactions can be neglected, and this requires that the dispersed second phase occupies a low volume fraction, even though high mass loading ($\dot{m}_{particles} \geq \dot{m}_{fluid}$) is acceptable. The particle or droplet trajectories are computed individually at specified intervals during the fluid phase calculation. This makes the model appropriate for the modeling of spray dryers, coal and liquid fuel combustion, and some particle-laden flows, but inappropriate for the modeling of liquid-liquid mixtures, fluidized beds, or any application where the volume fraction of the second phase cannot be neglected. For applications such as these, particle-particle interactions can be included using the Discrete Element Model, which is discussed in Discrete Element Method Collision Model (p. 577).

Limitations of the discrete phase model are listed in Limitations in the User's Guide.

12.2. Particle Motion Theory

This section is composed of the following:

12.2.1. Equations of Motion for Particles
12.2.2. Turbulent Dispersion of Particles
12.2.3. Integration of Particle Equation of Motion

12.2.1. Equations of Motion for Particles

12.2.1.1. Particle Force Balance

Ansys Fluent predicts the trajectory of a discrete phase particle (or droplet or bubble) by integrating the force balance on the particle, which is written in a Lagrangian reference frame. This force balance equates the particle inertia with the forces acting on the particle, and can be written as

$$m_p \frac{d\vec{u}_p}{dt} = m_p \frac{\vec{u} - \vec{u}_p}{\tau_r} + m_p \frac{\vec{g}\left(\rho_p - \rho\right)}{\rho_p} + \vec{F} \tag{12.1}$$

where m_p is the particle mass, $\vec{u}$ is the fluid phase velocity, $\vec{u}_p$ is the particle velocity, ρ is the fluid density, ρ_p is the density of the particle, $\vec{F}$ is an additional force, $m_p \frac{\vec{u} - \vec{u}_p}{\tau_r}$ is the drag force, and τ_r is the droplet or particle relaxation time [215] (p. 1065) calculated by:

$$\tau_r = \frac{\rho_p d_p^2}{18\mu} \frac{24}{C_d \mathrm{Re}} \tag{12.2}$$

Here, μ is the molecular viscosity of the fluid, d_p is the particle diameter, and Re is the relative Reynolds number, which is defined as

$$Re \equiv \frac{\rho d_p \left|\vec{u}_p - \vec{u}\right|}{\mu} \tag{12.3}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

12.2.1.2. Particle Torque Balance

Particle rotation is a natural part of particle motion and can have a significant influence on the trajectory of a particle moving in a fluid. The impact is even more pronounced for large and/or heavy particles with high moments of inertia. In this case, if particle rotation is disregarded in simulation studies, the resulting particle trajectories can significantly differ from the actual particle paths. To account for particle rotation, an additional ordinary differential equation (ODE) for the particle's angular momentum is solved:

$$I_p \frac{d\vec{\omega}_p}{dt} = \frac{\rho_f}{2}\left(\frac{d_p}{2}\right)^5 C_\omega |\vec{\Omega}| \cdot \vec{\Omega} = \vec{T} \tag{12.4}$$

where I_p is the moment of inertia, $\vec{\omega}_p$ is the particle angular velocity, ρ_f is the fluid density, d_p is the particle diameter, C_ω is the rotational drag coefficient, $\vec{T}$ is the torque applied to a particle in a fluid domain, and $\vec{\Omega}$ is the relative particle–fluid angular velocity calculated by:

$$\vec{\Omega} = \frac{1}{2}\nabla \times \vec{u}_f - \vec{\omega}_p \tag{12.5}$$

For a spherical particle, the moment of inertia I_p is calculated as:

$$I_p = \frac{\pi}{60}\rho_p d_p^5 \tag{12.6}$$

From Equation 12.4 (p. 449), it is apparent that the torque $\vec{T}$ results from equilibrium between the particle inertia and the drag.

For details on how to use the particle rotation capability, see Particle Rotation in the *Fluent User's Guide*.

12.2.1.3. Inclusion of the Gravity Term

While Equation 12.1 (p. 448) includes a force of gravity on the particle, it is important to note that in Ansys Fluent the default gravitational acceleration is zero. If you want to include the gravitational force, you must remember to define the magnitude and direction of the gravity vector in the Operating Conditions Dialog Box.

12.2.1.4. Other Forces

Equation 12.1 (p. 448) incorporates additional forces ($\vec{F}$) in the particle force balance that can be important under special circumstances. The first of these is the "virtual mass" force, the force required to accelerate the fluid surrounding the particle. This force can be written as

$$\vec{F} = C_{vm} m_p \frac{\rho}{\rho_p}\left(\vec{u}_p \nabla \vec{u} - \frac{d\vec{u}_p}{dt}\right) \tag{12.7}$$

where C_{vm} is the virtual mass factor with a default value of `0.5`.

An additional force arises due to the pressure gradient in the fluid:

$$\vec{F} = m_p \frac{\rho}{\rho_p} \vec{u} \nabla \vec{u} \tag{12.8}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

The virtual mass and pressure gradient forces are not important when the density of the fluid is much lower than the density of the particles as is the case for liquid/solid particles in gaseous flows ($\rho/\rho_p \ll 1$). For values of ρ/ρ_p approaching unity, the Virtual Mass and Pressure Gradient forces become significant and it is recommended that they be included when the density ratio is greater than 0.1. See Including the Virtual Mass Force and Pressure Gradient Effects on Particles in Fluent User's Guide for details on how to include these forces in your model.

12.2.1.5. Forces in Moving Reference Frames

The additional force term, $\vec{F}$, in Equation 12.1 (p. 448) also includes forces on particles that arise due to rotation of the reference frame. These forces arise when you are modeling flows in moving frames of reference (see Flow in a Moving Reference Frame (p. 21)). For rotation defined about the z axis, for example, the forces on the particles in the Cartesian x and y directions can be written as

$$m_p\left(1-\frac{\rho}{\rho_p}\right)\Omega^2 x+2m_p\Omega\left(u_{p,y}-\frac{\rho}{\rho_p}u_y\right) \tag{12.9}$$

where $u_{p,y}$ and u_y are the particle and fluid velocities in the Cartesian y direction, Ω is the RPM, and

$$m_p\left(1-\frac{\rho}{\rho_p}\right)\Omega^2 y-2m_p\Omega\left(u_{p,x}-\frac{\rho}{\rho_p}u_x\right) \tag{12.10}$$

where $u_{p,x}$ and u_x are the particle and fluid velocities in the Cartesian x direction.

12.2.1.6. Thermophoretic Force

Small particles suspended in a gas that has a temperature gradient experience a force in the direction opposite to that of the gradient. This phenomenon is known as thermophoresis. Ansys Fluent can optionally include a thermophoretic effect on particles in the additional force F in Equation 12.1 (p. 448):

$$\vec{F}=-D_{T,p}\frac{1}{T}\nabla T \tag{12.11}$$

where $D_{T,p}$ is the thermophoretic coefficient. You can define the coefficient to be constant, polynomial, or a user-defined function, or you can use the form suggested by Talbot [641] (p. 1090):

$$D_{T,p}=\frac{6\pi d_p\mu^2 C_s(K+C_tKn)}{\rho(1+3C_mKn)(1+2K+2C_tKn)} \tag{12.12}$$

where:

Kn = Knudsen number = 2 $\lambda/\ d_p$

λ = mean free path of the fluid

$K = k/\ k_p$

k = fluid thermal conductivity based on translational energy only = (15/4) μR

k_p = particle thermal conductivity

C_S = 1.17

C_t = 2.18

C_m = 1.14

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

T = local fluid temperature

μ = fluid viscosity

This expression assumes that the particle is a sphere and that the fluid is an ideal gas.

12.2.1.7. Brownian Force

For sub-micron particles, the effects of Brownian motion can be optionally included in the additional force term. The components of the Brownian force are modeled as a Gaussian white noise process with spectral intensity $S_{n,ij}$ given by [361] (p. 1074):

$$S_{n,ij}=S_0\delta_{ij} \tag{12.13}$$

where δ_{ij} is the Kronecker delta function, and

$$S_0=\frac{216\nu k_B T}{\pi^2\rho d_p^5\left(\frac{\rho_p}{\rho}\right)^2 C_c} \tag{12.14}$$

T is the absolute temperature of the fluid, ν is the kinematic viscosity, C_c is the Cunningham correction (defined in Equation 12.55 (p. 459)), and k_B is the Boltzmann constant. Amplitudes of the Brownian force components are of the form

$$F_{b_i}=m_p\zeta_i\sqrt{\frac{\pi S_o}{\Delta t}} \tag{12.15}$$

where ζ_i are zero-mean, unit-variance-independent Gaussian random numbers. The amplitudes of the Brownian force components are evaluated at each time step. The energy equation must be enabled in order for the Brownian force to take effect. Brownian force is intended only for laminar simulations.

12.2.1.8. Saffman's Lift Force

The Saffman's lift force, or lift due to shear, can also be included in the additional force term as an option. The lift force used is from Li and Ahmadi [361] (p. 1074) and is a generalization of the expression provided by Saffman [559] (p. 1085):

$$\vec{F}=m_p\frac{2K\nu^{1/2}\rho d_{ij}}{\rho_p d_p\left(d_{lk}d_{kl}\right)^{1/4}}\left(\vec{u}-\vec{u}_p\right) \tag{12.16}$$

where $K=2.594$ and d_{ij} is the deformation tensor. This form of the lift force is intended for small particle Reynolds numbers and is recommended only for sub-micron particles.

12.2.1.9. Magnus Lift Force

The Magnus or rotational lift force arises when particle is rotating in a fluid. The lift is caused by a pressure differential along a particle's surface ([124] (p. 1060)). For high Reynolds numbers, the Magnus force F_{RL} is scaled by a rotational lift coefficient C_{RL}:

$$F_{RL}=\frac{1}{2}A_p C_{RL}\rho_f\frac{|\vec{V}|}{|\vec{\Omega}|}\left(\vec{V}\times\vec{\Omega}\right) \tag{12.17}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

where,

A_p = projected particle surface area

$\vec{V}$ = relative fluid - particle velocity

$\vec{\Omega}$ = relative fluid-particle angular velocity

For the rotational lift coefficient C_{RL}, different approaches are available in the literature. Ansys Fluent offers the following formulations:

- Oesterle and Bui Dinh ([481] (p. 1081))

  The rotational lift coefficient C_{RL} is dependent on both the rotational Reynolds number Re_ω and the particle Reynolds number Re_p:

  $$C_{RL}=0.45+\left(4\frac{Re_\omega}{Re_p}-0.45\right)exp\left(-0.09896Re_\omega^{0.4}Re_p^{0.3}\right) \tag{12.18}$$

  The correlation provided by the authors is confirmed by their experimental measurements for Re_p up to 140. Comparison with experimental data provided by other authors suggests also that the correlation has a validity for Re_pup to 2000.

- Tsuji et al. ([657] (p. 1091))

  The rotational lift coefficient C_{RL} is defined as a function of the spin parameter S as follows:

  $$\begin{aligned} C_{RL}&=0.4 && for\,S\geq 1\\ C_{RL}&=(0.4\pm 0.1)S && for\,S<1 \end{aligned} \tag{12.19}$$

  The spin parameter S is defined as:

  $$S=\frac{|\vec{\omega}_p|d_p}{2|\vec{u}_f-\vec{u}_p|} \tag{12.20}$$

  This formulation is widely used with validity up to Re_p < 1600.

- Rubinow and Keller ([558] (p. 1085))

  In this primary assumption, the rotational lift coefficient C_{RL} is linearly proportional to the spin parameter S:

  $$C_{RL}=2S \tag{12.21}$$

  This model could be useful for comparison purposes or in academic applications.

For rotating particle simulations, you can include the Magnus lift force in the additional force term in the particle force balance equation (Equation 12.1 (p. 448)). For details, see Particle Rotation in the *Fluent User's Guide*.

12.2.2. Turbulent Dispersion of Particles

The dispersion of particles due to turbulence in the fluid phase can be predicted using the stochastic tracking model. The stochastic tracking (random walk) model includes the effect of instantaneous turbulent velocity fluctuations on the particle trajectories through the use of stochastic methods (see

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Stochastic Tracking (p. 453)). For stochastic tracking a model is available to account for the generation or dissipation of turbulence in the continuous phase (see Coupling Between the Discrete and Continuous Phases (p. 583)).

Important:

Turbulent dispersion of particles cannot be included if the Spalart-Allmaras turbulence model is used.

12.2.2.1. Stochastic Tracking

When the flow is turbulent, Ansys Fluent will predict the trajectories of particles using the mean fluid phase velocity, $\overline{u}$, in the trajectory equations (Equation 12.1 (p. 448)). Optionally, you can include the instantaneous value of the fluctuating gas flow velocity,

$$u=\overline{u}+u' \tag{12.22}$$

to predict the dispersion of the particles due to turbulence.

In the stochastic tracking approach, Ansys Fluent predicts the turbulent dispersion of particles by integrating the trajectory equations for individual particles, using the instantaneous fluid velocity, $\overline{u}+u'(t)$, along the particle path during the integration. By computing the trajectory in this manner for a sufficient number of representative particles (termed the "number of tries"), the random effects of turbulence on the particle dispersion can be included.

Ansys Fluent uses a stochastic method (random walk model) to determine the instantaneous gas velocity. In the discrete random walk (DRW) model, the fluctuating velocity components are discrete piecewise constant functions of time. Their random value is kept constant over an interval of time given by the characteristic lifetime of the eddies.

The DRW model may give nonphysical results in strongly nonhomogeneous diffusion-dominated flows, where small particles should become uniformly distributed. Instead, the DRW will show a tendency for such particles to concentrate in low-turbulence regions of the flow. In addition, this model is known to give poor prediction of the wall impaction rate of particles with a diameter less than a few microns due to turbulence.

12.2.2.1.1. The Integral Time

Prediction of particle dispersion makes use of the concept of the integral time scale, T:

$$T=\int_0^\infty \frac{u_p'(t)u_p'(t-\tau)}{\overline{u_p'^2}}d\tau \tag{12.23}$$

The integral time is proportional to the particle dispersion rate, as larger values indicate more turbulent motion in the flow. It can be shown that the particle diffusivity is given by $\overline{u_i'u_j'}T$.

For small "tracer" particles that move with the fluid (zero drift velocity), the integral time becomes the fluid Lagrangian integral time, T_L. This time scale can be approximated as

$$T_L=C_L\frac{k}{\varepsilon} \tag{12.24}$$

where C_L is to be determined as it is not well known. By matching the diffusivity of tracer particles, $\overline{u_i' u_j'} T_L$, to the scalar diffusion rate predicted by the turbulence model, ν_t / σ, one can obtain

$$T_L \approx 0.15 \frac{k}{\varepsilon} \tag{12.25}$$

for the k- ε model and its variants, and

$$T_L \approx 0.30 \frac{k}{\varepsilon} \tag{12.26}$$

when the Reynolds stress model (RSM) is used [132] (p. 1060). For the k- ω models, substitute $\varepsilon = 0.09 k\omega$ into Equation 12.24 (p. 453). The LES model uses the equivalent LES time scales.

12.2.2.1.2. The Discrete Random Walk Model

In the discrete random walk (DRW) model, or "eddy lifetime" model, the interaction of a particle with a succession of discrete stylized fluid phase turbulent eddies is simulated [215] (p. 1065). Each eddy is characterized by

- a Gaussian distributed random velocity fluctuation, u', v', and w'
- a time scale, τ_e

The values of u', v', and w' that prevail during the lifetime of the turbulent eddy are sampled by assuming that they obey a Gaussian probability distribution, so that

$$u' = \zeta \sqrt{\overline{u'^2}} \tag{12.27}$$

where ζ is a normally distributed random number, and the remainder of the right-hand side is the local RMS value of the velocity fluctuations. Since the kinetic energy of turbulence is known at each point in the flow, these values of the RMS fluctuating components can be defined (assuming isotropy) as

$$\sqrt{\overline{u'^2}} = \sqrt{\overline{v'^2}} = \sqrt{\overline{w'^2}} = \sqrt{2k / 3} \tag{12.28}$$

for the k- ε model, the k- ω model, and their variants. When the RSM is used, nonisotropy of the stresses is included in the derivation of the velocity fluctuations:

$$u' = \zeta \sqrt{\overline{u'^2}} \tag{12.29}$$

$$v' = \zeta \sqrt{\overline{v'^2}} \tag{12.30}$$

$$w' = \zeta \sqrt{\overline{w'^2}} \tag{12.31}$$

when viewed in a reference frame in which the second moment of the turbulence is diagonal [728] (p. 1095). For the LES model, the velocity fluctuations are equivalent in all directions. See Inlet Boundary Conditions for Scale Resolving Simulations (p. 126) for details.

The characteristic lifetime of the eddy is defined either as a constant:

$$\tau_e = 2T_L \tag{12.32}$$

where T_L is given by Equation 12.24 (p. 453) in general (Equation 12.25 (p. 454) by default), or as a random variation about T_L:

$$\tau_e = -T_L \ln(r) \tag{12.33}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

where r is a uniform random number greater than zero and less than 1 and T_L is given by Equation 12.25 (p. 454). The option of random calculation of τ_e yields a more realistic description of the correlation function.

The particle eddy crossing time is defined as

$$t_{cross} = -\tau \ln\left[1 - \left(\frac{L_e}{\tau|u - u_p|}\right)\right] \tag{12.34}$$

where τ is the particle relaxation time, L_e is the eddy length scale, and $|u - u_p|$ is the magnitude of the relative velocity.

The particle is assumed to interact with the fluid phase eddy over the smaller of the eddy lifetime and the eddy crossing time. When this time is reached, a new value of the instantaneous velocity is obtained by applying a new value of ζ in Equation 12.27 (p. 454).

12.2.2.1.3. Using the DRW Model

The only inputs required for the DRW model are the value for the integral time-scale constant, C_L (see Equation 12.24 (p. 453) and Equation 12.32 (p. 454)) and the choice of the method used for the prediction of the eddy lifetime. You can choose to use either a constant value or a random value by selecting the appropriate option in the Set Injection Properties Dialog Box for each injection, as described in Stochastic Tracking in the User's Guide.

Important:

Turbulent dispersion of particles cannot be included if the Spalart-Allmaras turbulence model is used.

12.2.3. Integration of Particle Equation of Motion

The trajectory equations, and any auxiliary equations describing heat or mass transfer to/from the particle, are solved by stepwise integration over discrete time steps. Integration of time in Equation 12.1 (p. 448) yields the velocity of the particle at each point along the trajectory, with the trajectory itself predicted by

$$\frac{dx}{dt} = u_p \tag{12.35}$$

Note that Equation 12.1 (p. 448) and Equation 12.35 (p. 455) are a set of coupled ordinary differential equations, and Equation 12.1 (p. 448) can be cast into the following general form

$$\frac{du_p}{dt} = \frac{1}{\tau_p}\left(u - u_p\right) + a \tag{12.36}$$

where the term a includes accelerations due to all other forces except drag force.

This set can be solved for constant u, a and τ_p by analytical integration. For the particle velocity at the new location u_p^{n+1} we get

$$u_p^{n+1} = u^n + e^{-\frac{\Delta t}{\tau_p}}\left(u_p^n - u^n\right) - a\tau_p\left(e^{-\frac{\Delta t}{\tau_p}} - 1\right) \tag{12.37}$$

The new location x_p^{n+1} can be computed from a similar relationship.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$x_p^{n+1}=x_p^n+\Delta t\left(u^n+a\tau_p\right)+\tau_p\left(1-e^{-\frac{\Delta t}{\tau_p}}\right)\left(u_p^n-u^n-a\tau_p\right) \tag{12.38}$$

In these equations u_p^n and u^n represent particle velocities and fluid velocities at the old location. Equation 12.37 (p. 455) and Equation 12.38 (p. 456) are applied when using the **analytic** discretization scheme.

The set of Equation 12.1 (p. 448) and Equation 12.35 (p. 455) can also be solved using numerical discretization schemes. When applying the Euler implicit discretization to Equation 12.36 (p. 455) we get

$$u_p^{n+1}=\frac{u_p^n+\Delta t\left(a+\frac{u^n}{\tau_p}\right)}{1+\frac{\Delta t}{\tau_p}} \tag{12.39}$$

When applying a trapezoidal discretization to Equation 12.36 (p. 455) the variables u_p and u_n on the right hand side are taken as averages, while accelerations, a, due to other forces are held constant. We get

$$\frac{u_p^{n+1}-u_p^n}{\Delta t}=\frac{1}{\tau_p}\left(u^*-u_p^*\right)+a^n \tag{12.40}$$

The averages u_p^* and u^* are computed from

$$u_p^*=\frac{1}{2}\left(u_p^n+u_p^{n+1}\right) \tag{12.41}$$

$$u^*=\frac{1}{2}\left(u^n+u^{n+1}\right) \tag{12.42}$$

$$u^{n+1}=u^n+\Delta t u_p^n\cdot\nabla u^n \tag{12.43}$$

The particle velocity at the new location $n+1$ is computed by

$$u_p^{n+1}=\frac{u_p^n\left(1-\frac{1}{2}\frac{\Delta t}{\tau_p}\right)+\frac{\Delta t}{\tau_p}\left(u^n+\frac{1}{2}\Delta t u_p^n\cdot\nabla u^n\right)+\Delta t a^n}{1+\frac{1}{2}\frac{\Delta t}{\tau_p}} \tag{12.44}$$

For the **implicit** and the **trapezoidal** schemes the new particle location is always computed by a trapezoidal discretization of Equation 12.35 (p. 455).

$$x_p^{n+1}=x_p^n+\frac{1}{2}\Delta t\left(u_p^n+u_p^{n+1}\right) \tag{12.45}$$

Equation 12.36 (p. 455) and Equation 12.35 (p. 455) can also be computed using a Runge-Kutta scheme which was published by Cash and Karp [95] (p. 1058). The ordinary differential equations can be considered as vectors, where the left hand side is the derivative $\vec{y}'$ and the right hand side is an arbitrary function $\vec{f}\left(t,\vec{y}\right)$.

$$\vec{y}'=\vec{f}\left(t,\vec{y}\right) \tag{12.46}$$

We get

$$\vec{y}^{n+1}=\vec{y}^n+c_1\vec{k}_1+c_2\vec{k}_2+c_3\vec{k}_3+c_4\vec{k}_4+c_5\vec{k}_5+c_6\vec{k}_6 \tag{12.47}$$

with

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$
\begin{aligned}
\vec{k}_1 &= \Delta t \vec{f}\left(t, \vec{y}^n\right) \\
\vec{k}_2 &= \Delta t \vec{f}\left(t + a_2 \Delta t, \vec{y}^n + b_{21}\vec{k}_1\right) \\
\vec{k}_3 &= \Delta t \vec{f}\left(t + a_3 \Delta t, \vec{y}^n + b_{31}\vec{k}_1 + b_{32}\vec{k}_2\right) \\
\vec{k}_4 &= \Delta t \vec{f}\left(t + a_4 \Delta t, \vec{y}^n + b_{41}\vec{k}_1 + b_{42}\vec{k}_2 + b_{43}\vec{k}_3\right) \\
\vec{k}_5 &= \Delta t \vec{f}\left(t + a_5 \Delta t, \vec{y}^n + b_{51}\vec{k}_1 + b_{52}\vec{k}_2 + b_{53}\vec{k}_3 + b_{54}\vec{k}_4\right) \\
\vec{k}_6 &= \Delta t \vec{f}\left(t + a_6 \Delta t, \vec{y}^n + b_{61}\vec{k}_1 + b_{62}\vec{k}_2 + b_{63}\vec{k}_3 + b_{64}\vec{k}_4 + b_{65}\vec{k}_5\right)
\end{aligned}
\tag{12.48}
$$

The coefficients $a_2 ... a_6$, $b_{21} ... b_{65}$, and $c_1 ... c_6$ are taken from Cash and Karp [95] (p. 1058).

This scheme provides an embedded error control, which is switched off, when no **Accuracy Control** is enabled.

For moving reference frames, the integration is carried out in the moving frame with the extra terms described in Equation 12.9 (p. 450) and Equation 12.10 (p. 450), therefore accounting for system rotation. Using the mechanisms available for accuracy control, the trajectory integration will be done accurately in time.

The **analytic** scheme is very efficient. It can become inaccurate for large steps and in situations where the particles are not in hydrodynamic equilibrium with the continuous flow. The numerical schemes **implicit** and **trapezoidal**, in combination with **Automated Tracking Scheme Selection**, consider most of the changes in the forces acting on the particles and are chosen as default schemes. The **runge-kutta** scheme is recommended if the non-drag force changes along a particle integration step.

The integration step size of the higher-order schemes, **trapezoidal** and **runge-kutta**, is limited to a stable range based on the particle momentum response time. Therefore it is recommended that you use them in combination with **Automated Tracking Scheme Selection**.

For the massless particle type, the particle velocity is equal to the velocity of the continuous phase, hence the solution of only the trajectory Equation 12.35 (p. 455) is required where the particle velocity $u_p = u$. The new particle location along the trajectory is always computed by Equation 12.41 (p. 456) and Equation 12.45 (p. 456), with $u_p = u$.

The equation for rotating particle motion (Equation 12.195 (p. 493)) is solved using the Euler implicit discretization scheme.

12.3. Laws for Drag Coefficients

Several laws for drag coefficients, C_D, are available for the Euler-Lagrange Model.

12.3.1. Spherical Drag Law

12.3.2. Non-spherical Drag Law

12.3.3. Stokes-Cunningham Drag Law

12.3.4. High-Mach-Number Drag Law

12.3.5. Dynamic Drag Model Theory

12.3.6. Dense Discrete Phase Model Drag Laws

12.3.7. Bubbly Flow Drag Laws

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

12.3.8. Rotational Drag Law

Instructions for selecting the drag law are provided in Drag Laws in the User's Guide.

12.3.1. Spherical Drag Law

The drag coefficient, C_D, for smooth particles can be taken from

$$C_D = a_1 + \frac{a_2}{Re} + \frac{a_3}{Re^2} \tag{12.49}$$

where a_1, a_2, and a_3 are constants that apply over several ranges of Re given by Morsi and Alexander [458] (p. 1079).

12.3.2. Non-spherical Drag Law

For non-spherical particles Haider and Levenspiel [225] (p. 1066) developed the correlation

$$C_D = \frac{24}{Re_{sph}}\left(1 + b_1 {Re_{sph}}^{b_2}\right) + \frac{b_3 Re_{sph}}{b_4 + Re_{sph}} \tag{12.50}$$

where

$$\begin{aligned} b_1 &= \exp\left(2.3288 - 6.4581\varphi + 2.4486\varphi^2\right) \\ b_2 &= 0.0964 + 0.5565\varphi \\ b_3 &= \exp\left(4.905 - 13.8944\varphi + 18.4222\varphi^2 - 10.2599\varphi^3\right) \\ b_4 &= \exp\left(1.4681 + 12.2584\varphi - 20.7322\varphi^2 + 15.8855\varphi^3\right) \end{aligned} \tag{12.51}$$

The shape factor, φ, is defined as

$$\varphi = \frac{s}{S} \tag{12.52}$$

where s is the surface area of a sphere having the same volume as the particle, and S is the actual surface area of the particle. For the purposes of calculating particle mass, drag force, and Re_{sph}, the particle size d_p should be the diameter of a sphere having the same volume.

Important:

The shape factor cannot exceed a value of 1.

12.3.3. Stokes-Cunningham Drag Law

The drag function F_D is expressed as:

$$F_D\left(u - u_p\right) = \frac{\mu}{\rho_p d_p^2}\frac{18 C_D Re}{24}\left(u - u_p\right) \tag{12.53}$$

where

μ = viscosity.

ρ_p = particle density

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

D_p = particle diameter

C_D = drag coefficient

Re = Reynolds number

For sub-micron particles, a form of Stokes' drag law more suitable for small particles is available [499] (p. 1082). In this case, F_D is defined as

$$F_D = \frac{18\mu}{d_p^2 \rho_p C_c} \tag{12.54}$$

The factor C_c is the Cunningham correction to Stokes' drag law, which you can compute from

$$C_c = 1 + \frac{2\lambda}{d_p}\left(1.257 + 0.4e^{-\left(1.1 d_p / 2\lambda\right)}\right) \tag{12.55}$$

where λ is the molecular mean free path. The Cunningham correction improves drag predictions for the slip flow or transitional regime between the continuum and free-molecular flow regimes. It is not intended for the free-molecular flow regime where the mean free path of the gas molecules is much larger than the particle diameter.

12.3.4. High-Mach-Number Drag Law

A high-Mach-number drag law is also available. This drag law is similar to the spherical law (Equation 12.49 (p. 458)) with corrections [111] (p. 1059) to account for a particle Mach number greater than 0.4 at a particle Reynolds number greater than 20.

12.3.5. Dynamic Drag Model Theory

Accurate determination of droplet drag coefficients is crucial for accurate spray modeling. Ansys Fluent provides a method that determines the droplet drag coefficient dynamically, accounting for variations in the droplet shape.

The dynamic drag model is applicable in almost any circumstance. It is compatible with both the TAB and wave models for droplet breakup. When the collision model is turned on, collisions reset the distortion and distortion velocities of the colliding droplets.

Many droplet drag models assume the droplet remains spherical throughout the domain. With this assumption, the drag of a spherical object is determined by the following [382] (p. 1075):

$$C_{d,sphere} = \begin{cases} 0.424 & Re > 1000 \\ \frac{24}{Re}\left(1 + \frac{1}{6}Re^{2/3}\right) & Re \leq 1000 \end{cases} \tag{12.56}$$

However, as an initially spherical droplet moves through a gas, its shape is distorted significantly when the Weber number is large. In the extreme case, the droplet shape will approach that of a disk. The drag of a disk, however, is significantly higher than that of a sphere. Since the droplet drag coefficient is highly dependent upon the droplet shape, a drag model that assumes the droplet is spherical is unsatisfactory. The dynamic drag model accounts for the effects of droplet distortion, linearly varying the drag between that of a sphere (Equation 12.56 (p. 459)) and a value of 1.54 corresponding to a disk [382] (p. 1075). The drag coefficient is given by

$$C_d = C_{d,sphere}\left(1 + 2.632y\right) \tag{12.57}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

where y is the droplet distortion, as determined by the solution of

$$\frac{d^2y}{dt^2} = \frac{C_F}{C_b}\frac{\rho_g}{\rho_l}\frac{u^2}{r^2} - \frac{C_k\sigma}{\rho_l r^3}y - \frac{C_d\mu_l}{\rho_l r^2}\frac{dy}{dt} \tag{12.58}$$

In the limit of no distortion ($y=0$), the drag coefficient of a sphere will be obtained, while at maximum distortion ($y=1$) the drag coefficient corresponding to a disk will be obtained.

Note that Equation 12.58 (p. 460) is obtained from the TAB model for spray breakup, described in Taylor Analogy Breakup (TAB) Model (p. 555), but the dynamic drag model can be used with either of the breakup models.

12.3.6. Dense Discrete Phase Model Drag Laws

The drag laws that are suitable for dense gas-solid flow are:

- Wen and Yu
- Gidaspow
- Syamlal-O'Brien
- Huilin-Gidaspow
- Gibilaro
- EMMS
- Filtered two-fluid model

A detailed theoretical background to the correlations, along with a recommendation of when to use each model, can be found in Fluid-Solid Exchange Coefficient (p. 675) (see also Specifying the Drag Function in the User's Guide). All these correlations incorporate group effects and, therefore, are dependent on the particle phase volume fraction. Thus, the drag models are only available when the DPM volume fraction is computed, that is, in conjunction with the Dense Discrete Phase Model.

12.3.7. Bubbly Flow Drag Laws

At sufficiently small particle Reynolds numbers (that is, gas bubbles and liquid droplets are very small), dispersed droplets and bubbles behave in the same manner as solid spherical particles. Hence, the drag coefficient is well approximated by the Schiller-Naumann or the Alexander-Morsi correlation.

At larger particle Reynolds numbers (the inertial or distorted particle regime), surface tension effects become important. The fluid particles become, at first, approximately ellipsoidal in shape, and finally, spherical cap shaped.

The following drag coefficient laws for sparsely distributed dispersed droplets and bubbles are available in Ansys Fluent:

- Ishii-Zuber Drag Model (p. 460) [272] (p. 1068)
- Grace Drag Model (p. 461) [111] (p. 1059)

12.3.7.1. Ishii-Zuber Drag Model

- Viscous Regime

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

In the viscous regime, dispersed droplets and bubbles behave in the same manner as solid spherical particles. Hence, the drag coefficient is well approximated by the Alexander-Morsi correlation:

$$C_{d,visc} = C_{d,sphere} \tag{12.59}$$

- Distorted Regime

  In the distorted particle regime, the drag coefficient is approximately constant, independent of Reynolds number, but dependent on particle shape through the dimensionless group known as the Eotvos number, which measures the ratio between gravitational and surface tension forces:

$$Eo = \frac{g\Delta\rho d_p^2}{\sigma} \tag{12.60}$$

  where $\Delta\rho$ is the density difference between the phases, g is the gravitational acceleration, and σ is the surface tension coefficient.

  For the distorted regime, the Ishii-Zuber correlation gives:

$$C_{d,dist} = \frac{2}{3}\sqrt{Eo} \tag{12.61}$$

- Spherical Cap Regime

  In the spherical cap regime, the drag coefficient is approximated by:

$$C_{d,cap} = \frac{8}{3} \tag{12.62}$$

  The bubble drag coefficient over a large particle Reynolds number range is computed as follows:

$$C_d = \max(C_{d,visc}, \min(C_{d,dist}, C_{d,cap})) \tag{12.63}$$

12.3.7.2. Grace Drag Model

- Viscous Regime

  In the viscous regime, the drag coefficient is modeled in the same way as in Ishii-Zuber Drag Model (p. 460) (Equation 12.59 (p. 461)).

- Distorted Regime

  The Grace drag model is formulated for flow past a single bubble. Here the drag coefficient in the distorted particle regime is given by:

$$C_{d,dist} = \frac{4}{3}\frac{gd_p}{u_t^2}\frac{\Delta\rho}{\rho_f} \tag{12.64}$$

  where u_t is the terminal velocity given by:

$$u_t = \frac{\mu_f}{\rho_f d_p} Mo^{(-0.149)}(J - 0.857) \tag{12.65}$$

  where Mo is the Morton number calculated by:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$Mo=\frac{\mu_f^4 g\Delta\rho}{\rho_f^2\sigma^3}$$

and

$$J=\begin{cases}0.94H^{0.757}, & if\ 2<H\leq 59.3\\ 3.42H^{0.441}, & if\ H>59.3\end{cases}$$

with

$$H=\frac{4}{3}EoMo^{-0.149}\left(\frac{\mu_f}{\mu_{\text{f,ref}}}\right)^{-0.14}$$

Here, $\mu_{\text{f,ref}}=9.0*10^{-4}\left[Pa\,s\right]$ is the molecular viscosity of water at 25°C and 1 bar.

In the above equations, subscript f refers to continuous phase.

- Spherical Cap Regime

  In the spherical cap regime, the drag coefficient is modeled in the same way as in Ishii-Zuber Drag Model (p. 460) (Equation 12.63 (p. 461)).

12.3.8. Rotational Drag Law

The rotational drag coefficient C_ω allows for higher rotational Reynolds numbers Re_ω, where Re_ω is defined as:

$$Re_\omega=\frac{\rho_f|\vec{\Omega}|d_p^2}{4\mu_f} \tag{12.66}$$

Ansys Fluent uses the correlation for the rotational drag coefficient C_ω proposed by Dennis et al. ([137] (p. 1060)):

$$C_\omega=\frac{6.45}{\sqrt{Re_\omega}}+\frac{32.1}{Re_\omega} \tag{12.67}$$

The range of validity for Equation 12.67 (p. 462) is: $20 \leq Re_\omega \leq 1000$.

12.4. Laws for Heat and Mass Exchange

Using Ansys Fluent's discrete phase modeling capability, reacting particles or droplets can be modeled and their impact on the continuous phase can be examined. Several heat and mass transfer relationships, termed "laws", are available in Ansys Fluent and the physical models employed in these laws are described in this section.

12.4.1. Inert Heating or Cooling (Law 1/Law 6)

12.4.2. Droplet Vaporization (Law 2)

12.4.3. Droplet Boiling (Law 3)

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

12.4.4. Devolatilization (Law 4)
12.4.5. Surface Combustion (Law 5)
12.4.6. Multicomponent Particle Definition (Law 7)

The laws that you activate depend upon the particle type that you select. In the Set Injection Properties Dialog Box you will specify the **Particle Type**, and Ansys Fluent will use a given set of heat and mass transfer laws for the chosen type. All particle types have predefined sequences of physical laws as shown in the table below:

Particle Type	Description	Laws Activated
Massless	–	–
Inert	inert/heating or cooling	1, 6
Droplet	heating/evaporation/boiling	1, 2, 3, 6
Combusting	heating; evolution of volatiles/swelling; heterogeneous surface reaction	1, 4, 5, 6
Multicomponent	multicomponent droplets/particles	7

In addition to the above laws, you can define your own laws using a user-defined function. More information about user-defined functions can be found in the Fluent Customization Manual.

You can also extend combusting particles to include an evaporating/boiling material by selecting **Wet Combustion** in the Set Injection Properties Dialog Box.

Ansys Fluent's physical laws (Laws 1 through 6), which describe the heat and mass transfer conditions listed in this table, are explained in detail in the sections that follow.

12.4.1. Inert Heating or Cooling (Law 1/Law 6)

The inert heating or cooling laws (Laws 1 and 6) are applied when the particle temperature is less than the vaporization temperature that you define, T_{vap}, and after the volatile fraction, $f_{v,0}$, of a particle has been consumed. These conditions may be written as

Law 1:

$$T_p < T_{vap} \tag{12.68}$$

Law 6:

$$m_p \leq \left(1 - f_{v,0}\right) m_{p,0} \tag{12.69}$$

where T_p is the particle temperature, $m_{p,0}$ is the initial mass of the particle, and m_p is its current mass.

Law 1 is applied until the temperature of the particle/droplet reaches the vaporization temperature. At this point a non-inert particle/droplet may proceed to obey one of the mass-transfer laws (2, 3, 4, and/or 5), returning to Law 6 when the volatile portion of the particle/droplet has been consumed. (Note that the vaporization temperature, T_{vap}, is an arbitrary modeling constant used to define the onset of the vaporization/boiling/volatilization laws.)

When using Law 1 or Law 6, Ansys Fluent uses a simple heat balance to relate the particle temperature, $T_p(t)$, to the convective heat transfer and the absorption/emission of radiation at the particle surface:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$m_p c_p \frac{dT_p}{dt} = hA_p\left(T_\infty - T_p\right) + \varepsilon_p A_p \sigma\left(\theta_R^4 - T_p^4\right) \tag{12.70}$$

where

m_p = mass of the particle (kg)

c_p = heat capacity of the particle (J/kg-K)

A_p = surface area of the particle (m^2)

T_∞ = local temperature of the continuous phase (K)

h = convective heat transfer coefficient ($\mathrm{W}/\mathrm{m}^2\text{-K}$)

ε_p = particle emissivity (dimensionless)

σ = Stefan-Boltzmann constant (5.67 x 10^{-8} $\mathrm{W/m^2\text{-}K^4}$)

θ_R = radiation temperature, $\left(\frac{G}{4\sigma}\right)^{1/4}$ (K)

Equation 12.70 (p. 464) assumes that there is negligible internal resistance to heat transfer, that is, the particle is at uniform temperature throughout.

G is the incident radiation in $\mathrm{W/m^2}$:

$$G = \int_{\Omega=4\pi} I d\Omega \tag{12.71}$$

where I is the radiation intensity and Ω is the solid angle.

Radiation heat transfer to the particle is included only if you have enabled the P-1 or discrete ordinates radiation heat transfer to particles using the **Particle Radiation Interaction** option in the Discrete Phase Model Dialog Box.

Equation 12.70 (p. 464) is integrated in time using an approximate, linearized form that assumes that the particle temperature changes slowly from one time value to the next:

$$m_p c_p \frac{dT_p}{dt} = A_p\left\{-\left[h + \varepsilon_p \sigma T_p^3\right] T_p + \left[hT_\infty + \varepsilon_p \sigma \theta_R^4\right]\right\} \tag{12.72}$$

As the particle trajectory is computed, Ansys Fluent integrates Equation 12.72 (p. 464) to obtain the particle temperature at the next time value, yielding

$$T_p\left(t + \Delta t\right) = \alpha_p + \left[T_p\left(t\right) - \alpha_p\right] e^{-\beta_p \Delta t} \tag{12.73}$$

where Δt is the integration time step and

$$\alpha_p = \frac{hT_\infty + \varepsilon_p \sigma \theta_R^4}{h + \varepsilon_p \sigma T_p^3\left(t\right)} \tag{12.74}$$

and

$$\beta_p = \frac{A_p\left(h + \varepsilon_p \sigma T_p^3\left(t\right)\right)}{m_p c_p} \tag{12.75}$$

Ansys Fluent can also solve Equation 12.72 (p. 464) in conjunction with the equivalent mass transfer equation using a stiff coupled solver. See Including Coupled Heat-Mass Solution Effects on the Particles in the User's Guide for details.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

The heat transfer coefficient, h, is evaluated using the correlation of Ranz and Marshall [539] (p. 1084), [540] (p. 1084):

$$Nu=\frac{hd_p}{k_\infty}=2.0+0.6Re_d^{1/2}Pr^{1/3} \tag{12.76}$$

where

d_p = particle diameter (m)

k_∞ = thermal conductivity of the continuous phase (W/m-K)

Re_d = Reynolds number based on the particle diameter and the relative velocity (Equation 12.3 (p. 448))

Pr = Prandtl number of the continuous phase ($c_p\mu/\ k_\infty$)

For inert and combusting particles, the following heat exchange coefficient correlations are available in Ansys Fluent:

- Constant HTC

  In this method, the volumetric heat transfer coefficient is specified as a constant value.
- Nusselt number

  In this method, the heat transfer coefficient is computed using the user specified Nusselt number Nu as follows:

  $$h=Nu\frac{k_\infty}{d_p} \tag{12.77}$$
- Tomiyama model

  See Tomiyama Model (p. 702) for details.
- Ranz-Marshall model (default)

  The model uses Equation 12.76 (p. 465) to evaluate the heat transfer coefficient.
- Hughmark model

  See Hughmark Model (p. 702) for details.
- Gunn model (DDPM only)

  See Gunn Model (p. 703) for details.

For more information on how to use these models, see Heat Transfer Coefficient in the *Fluent User's Guide*.

Finally, the heat lost or gained by the particle as it traverses each computational cell appears as a source or sink of heat in subsequent calculations of the continuous phase energy equation. During Laws 1 and 6, particles/droplets do not exchange mass with the continuous phase and do not participate in any chemical reaction.

12.4.2. Droplet Vaporization (Law 2)

Law 2 is applied to predict the vaporization from a discrete phase droplet. Law 2 is initiated when the temperature of the droplet reaches the vaporization temperature, T_{vap}, and continues until the droplet reaches the boiling point, T_{bp}, or until the droplet's volatile fraction is completely consumed:

$$T_{vap}\leq T_p<T_{bp} \tag{12.78}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$m_p > \left(1 - f_{v,0}\right) m_{p,0} \tag{12.79}$$

The onset of the vaporization law is determined by the setting of T_{vap}, a modeling parameter that has no physical significance. Note that once vaporization is initiated (by the droplet reaching this threshold temperature), it will continue to vaporize even if the droplet temperature falls below T_{vap}. Vaporization will be halted only if the droplet temperature falls below the dew point. In such cases, the droplet will remain in Law 2 but no evaporation will be predicted. When the boiling point is reached, the droplet vaporization is predicted by a boiling rate, Law 3, as described in a section that follows.

12.4.2.1. Mass Transfer During Law 2—Diffusion Controlled Model

During Law 2, when the rate of vaporization is slow, it can be assumed to be governed by gradient diffusion, with the flux of droplet vapor into the gas phase related to the difference in vapor concentration at the droplet surface and the bulk gas:

$$N_i = k_c \left(C_{i,s} - C_{i,\infty}\right) \tag{12.80}$$

where

N_i = molar flux of vapor $\left(\mathrm{kmol/m^2\text{-}s}\right)$

k_c = mass transfer coefficient $\left(\mathrm{m/s}\right)$

$C_{i,s}$ = vapor concentration at the droplet surface $\left(\mathrm{kmol/m^3}\right)$

$C_{i,\infty}$ = vapor concentration in the bulk gas $\left(\mathrm{kmol/m^3}\right)$

Note that Ansys Fluent's vaporization law assumes that N_i is positive (evaporation). If conditions exist in which N_i is negative (that is, the droplet temperature falls below the dew point and condensation conditions exist), Ansys Fluent treats the droplet as inert ($N_i = 0.0$).

The concentration of vapor at the droplet surface is evaluated by assuming that the partial pressure of vapor at the interface is equal to the saturated vapor pressure, p_{sat}, at the droplet temperature, T_p:

$$C_{i,s} = \frac{p_{sat}\left(T_p\right)}{RT_p} \tag{12.81}$$

where R is the universal gas constant.

The concentration of vapor in the bulk gas is known from solution of the transport equation for species i as:

$$C_{i,\infty} = X_i \frac{p}{RT_\infty} \tag{12.82}$$

where X_i is the local bulk mole fraction of species i, p is the local pressure, and T_∞ is the local bulk temperature in the gas. The mass transfer coefficient in Equation 12.80 (p. 466) is calculated from the Sherwood number correlation [539] (p. 1084), [540] (p. 1084):

$$Sh_{AB} = \frac{k_c d_p}{D_{i,m}} = 2.0 + 0.6 Re_d^{1/2} Sc^{1/3} \tag{12.83}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

where

$D_{i,m}$ = diffusion coefficient of vapor in the bulk $\left(\mathrm{m}^2/\mathrm{s}\right)$

Sc = the Schmidt number, $\mu/\rho D_{i,m}$

d_p = particle (droplet) diameter $\left(\mathrm{m}\right)$

The vapor flux given by Equation 12.80 (p. 466) becomes a source of species i in the gas phase species transport equation, (see Setting Material Properties for the Discrete Phase in the User's Guide) or in the mixture fraction equation for nonpremixed combustion calculations.

The mass of the droplet is reduced according to

$$m_p\left(t+\Delta t\right)=m_p\left(t\right)-N_i A_p M_{w,i}\Delta t \tag{12.84}$$

where

$M_{w,i}$ = molecular weight of species i $\left(\mathrm{kg}/\mathrm{kmol}\right)$

m_p = mass of the droplet $\left(\mathrm{kg}\right)$

A_p = surface area of the droplet (m^2)

Ansys Fluent can also solve Equation 12.84 (p. 467) in conjunction with the equivalent heat transfer equation using a stiff coupled solver. See Including Coupled Heat-Mass Solution Effects on the Particles in the User's Guide for details.

12.4.2.2. Mass Transfer During Law 2—Convection/Diffusion Controlled Model

For high vaporization rates, the effect of the convective flow of the evaporating material from the droplet surface to the bulk gas phase (Stefan Flow) becomes important.

In Ansys Fluent, the following expression has been adopted following the work of Miller [439] (p. 1078) and Sazhin [570] (p. 1086):

$$\frac{dm_p}{dt}=-k_c A_p \rho \ln\left(1+B_m\right) \tag{12.85}$$

where

m_p = droplet mass (kg)

k_c = mass transfer coefficient (m/s)

A_p = droplet surface area (m^2)

ρ = gas density ($\mathrm{kg/m}^3$)

B_m is the Spalding mass number given by:

$$B_m=\frac{Y_{i,s}-Y_{i,\infty}}{1-Y_{i,s}} \tag{12.86}$$

where

$Y_{i,s}$ = vapor mass fraction at the surface

$Y_{i,\infty}$ = vapor mass fraction in the bulk gas

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

The mass transfer coefficient k_c is given by Equation 12.83 (p. 466).

12.4.2.3. Mass Transfer During Law 2—Thermolysis

The single-rate thermolysis model [67] (p. 1056) uses the following Arrhenius expression to compute the mass transfer rate from the droplet to the bulk gas phase:

$$\frac{dm_p}{dt} = -\pi d_p A e^{-E/RT_p} \tag{12.87}$$

where

m_p = droplet mass (kg)

d_p = droplet diameter (m)

A = pre-exponential factor (kg/s-m)

E = activation energy (J/kg)

T_p = particle temperature (K)

A simplified constant rate thermolysis model is represented as:

$$\frac{dm_p}{dt} = -Cm_{p0} \tag{12.88}$$

where

C = rate constant (1/s)

m_{p0} = initial particle mass (kg)

12.4.2.4. Defining the Saturation Vapor Pressure and Diffusion Coefficient (or Binary Diffusivity)

You must define the saturation vapor pressure as a polynomial or piecewise linear function of temperature ($p_{sat}(T)$) during the problem definition. Note that the saturation vapor pressure definition is critical, as p_{sat} is used to obtain the driving force for the evaporation process (Equation 12.80 (p. 466) and Equation 12.81 (p. 466)). You should provide accurate vapor pressure values for temperatures over the entire range of possible droplet temperatures in your problem. Saturation vapor pressure data can be obtained from a physics or engineering handbook (for example, [508] (p. 1082)).

You must also input the diffusion coefficient, $D_{i,m}$, during the setup of the discrete phase material properties. Note that the diffusion coefficient inputs that you supply for the continuous phase are not used in the discrete phase model.

You can define the binary diffusion coefficient (or binary diffusivity) to be either constant, or a function of the continuous phase temperature. Alternatively, you can define the binary diffusion coefficient as function of a film-averaged temperature T_f, which is computed by

$$T_f = T_p + a(T_\infty - T_p) \tag{12.89}$$

where

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

T_p = droplet temperature (K)

T_∞ = temperature in the bulk gas (K)

a = averaging coefficient

You can also choose to have Fluent compute the diffusion coefficient based on the assumption of unity Lewis number:

$$Le_i = \frac{k}{\rho c_p D_{i,m}} = 1 \tag{12.90}$$

where

$D_{i,m}$ = mass diffusivity of species i in the gas-phase mixture

k = mixture thermal conductivity

ρ = mixture density

c_p = mixture specific heat

According to Polling et al. [522] (p. 1083) , for low to moderate pressures (for example, pressures < $0.9P_{critical}$), the binary diffusivity D_i is inversely proportional to the pressure P, so the following relation applies:

$$D_i = D_{i,ref} P_{ref} / P \tag{12.91}$$

where $D_{i,ref}$ is the binary diffusivity of species i at the reference pressure P_{ref}.

12.4.2.5. Defining the Boiling Point and Latent Heat

The boiling point T_{bp} and the latent heat h_{fg} are defined as constant property inputs for the droplet-particle materials. The default boiling point data in the Ansys Fluent property database correspond to a pressure of 1 atm (normal boiling point) and the latent heat data correspond to the normal boiling point of the droplets.

During the evaporation process, as the particle changes its temperature, the latent heat will vary according to Equation 12.92 (p. 469)

$$h_{fg} = -\int_{T_p}^{T_{bp}} c_{p,g}\, dT + h_{fg,bp} + \int_{T_p}^{T_{bp}} c_{p,p}\, dT \tag{12.92}$$

where

T_{bp} = boiling point (K)

$h_{\mathrm{fg,bp}}$ = latent heat at the boiling point (J/kg)

$c_{\mathrm{p,g}}$ and $c_{\mathrm{p,p}}$ = specific heat of gaseous and liquid phase, respectively (J/kg/K)

For simulations at or near atmospheric pressure, the latent heat variation with droplet temperature is generally small and can be ignored, so $h_{fg} \approx h_{fg,bp}$. To include the droplet temperature effects on the latent heat according to Equation 12.92 (p. 469), see Including the Effect of Droplet Temperature on Latent Heat in the *Fluent User's Guide*.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

If the pressure in your simulation differs from atmospheric, you will have to revise the default boiling point data in the Ansys Fluent property database. Refer to Considering Pressure Dependence in Boiling in the *Fluent User's Guide* for details.

When the Real Gas Model (RGM) (see Real Gas Models in the *Fluent User's Guide*) is used in a DPM simulation, the appropriate modeling approach depends on the operating condition regimes. Refer to Using the Cubic Equation of State Models with the Lagrangian Dispersed Phase Models in the *Fluent User's Guide* for guidelines and restrictions of this approach. Note that the NIST RGM is not compatible with the DPM, except for the inert and massless particle types.

12.4.2.6. Heat Transfer to the Droplet

Finally, the droplet temperature is updated according to a heat balance that relates the sensible heat change in the droplet to the convective and latent heat transfer between the droplet and the continuous phase:

$$m_p c_p \frac{dT_p}{dt} = hA_p\left(T_\infty - T_p\right) + \frac{dm_p}{dt} h_{fg} + A_p \varepsilon_p \sigma\left(\theta_R^4 - T_p^4\right) \tag{12.93}$$

where:

c_p = droplet heat capacity (J/kg-K)

T_p = droplet temperature (K)

h = convective heat transfer coefficient $\left(\mathrm{W/m^2\text{-}K}\right)$

T_∞ = temperature of continuous phase (K)

dm_p/dt = rate of change of particle mass during evaporation (kg/s) (negative value)

h_{fg} = latent heat (J/kg)

ε_p = particle emissivity (dimensionless)

σ = Stefan-Boltzmann constant $\left(5.67\times10^{-8}\mathrm{W/m^2\text{-}K^4}\right)$

θ_R = radiation temperature, $\left(\frac{G}{4\sigma}\right)^{1/4}$, where G is the incident radiation defined by Equation 12.71 (p. 464)

Radiation heat transfer to the particle is included only if you have enabled the P-1 or discrete ordinates radiation heat transfer to particles using the **Particle Radiation Interaction** option in the Discrete Phase Model Dialog Box.

The heat transferred to or from the gas phase becomes a source/sink of energy during subsequent calculations of the continuous phase energy equation.

When the vaporization rate is computed by the Convection/Diffusion Controlled model (Equation 12.85 (p. 467)), the convective heat transfer coefficient h in Equation 12.93 (p. 470) is calculated with a modified Nu number as follows [570] (p. 1086):

$$Nu = \frac{hd_p}{k} = \frac{\ln\left(1+B_T\right)}{B_T}\left(2+0.6\mathrm{Re}_d^{1/2}\mathrm{Pr}^{1/3}\right) \tag{12.94}$$

where

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

d_p = particle diameter (m)

k = thermal conductivity of the continuous phase (W/m-K)

Re_d = Reynolds number (Equation 12.3 (p. 448))

Pr = Prandtl number of the continuous phase ($c_p\mu/k$)

B_T is the Spalding heat transfer number defined as:

$$B_T = \frac{c_{pv}\left(T_\infty - T_p\right)}{h_{fg} - \dot{q}_p/\dot{m}_p} \tag{12.95}$$

where

$\dot{q}_p$ = heat transferred to the droplet (w)

$\dot{m}_p$ = particle evaporation rate (kg/s)

c_{pv} = specific heat of droplet vapor (J/kg-K)

B_T is computed from the Spalding mass number by:

$$B_T = \left(1 + B_m\right)^{\frac{1}{Le}\cdot\frac{Sh}{Nu}\cdot\frac{c_{pv}}{c_{pg}}} - 1 \tag{12.96}$$

where

Le = the Lewis number = $\frac{k}{c_{pg}\rho D_{i,m}}$

c_{pg} = specific heat of gas mixture (J/kg-K)

For the unity Lewis number and with the further assumption that c_{pv}=c_{pg}, Equation 12.96 (p. 471) reduces to:

$$B_T = B_m \tag{12.97}$$

When temperature differences between droplet and bulk gas are large, transient effects become important. The assumption of uniform droplet temperature and the assumption of emanating vapor in temperature equilibrium with the bulk gas may be questioned. In such conditions, an averaging of the Spalding heat transfer term may produce more realistic results.

ANSYS Fluent provides the option to average the term $\ln\left(1+B_T\right)/B_T$ in Equation 12.94 (p. 470) with the evaporating species surface mass fraction as follows:

$$F_T = 1 - Y_{is}\left(1 - F_{T_0}\right) \tag{12.98}$$

where

$F_{T_0} = \ln\left(1+B_T\right)/B_T$

Y_{is} = surface mass fraction of the evaporating species

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

12.4.3. Droplet Boiling (Law 3)

Law 3 is applied to predict the convective boiling of a discrete phase droplet when the temperature of the droplet has reached the boiling temperature, T_{bp}, and while the particle still contains mass that can evaporate (the particle mass is larger than the non-volatile mass, $\left(1-f_{v,0}\right)m_{p,0}$):

$$T_p \geq T_{bp} \tag{12.99}$$

and

$$m_p > \left(1-f_{v,0}\right)m_{p,0} \tag{12.100}$$

When the droplet temperature reaches the boiling point T_{bp}, the boiling rate is computed by solving equations Equation 12.93 (p. 470) and Equation 12.94 (p. 470) for the particle diameter to yield:

$$\frac{d\left(d_p\right)}{dt} = -\frac{2k_\infty}{\rho_p c_{p\infty} d_p}\left(2+0.6Re^{1/2}Pr^{1/3}\right)\ln\left[1+\frac{c_{p\infty}\left(T_\infty - T_p\right)}{h_{fg}}\right] \tag{12.101}$$

where

$c_{p,\infty}$ = heat capacity of the gas (J/kg-K)

ρ_p = droplet density (kg/m^3)

Re_d = Reynolds number

Pr = Prandtl number of the continuous phase

k_∞ = thermal conductivity of the gas (W/m-K)

Note that pre-2019 R3 releases of Ansys Fluent used a simplified version of Equation 12.101 (p. 472) in which the Prandtl number was assumed constant and equal to 0.45. In this case, the term $0.6Re^{1/2}Pr^{1/3}$ reduces to $0.46Re^{1/2}$.

Ansys Fluent provides the option to set the heat capacity of the gas $c_{p,\infty}$ equal to the heat capacity of the evaporating species. See Description of the Properties in the *Fluent User's Guide* for more information about this option.

Equation 12.101 (p. 472) was derived assuming steady flow at constant pressure. Note that the model requires $T_\infty > T_{boil}$ in order for boiling to occur and that the droplet remains at fixed temperature (T_{boil}) throughout the boiling law.

When radiation heat transfer is active, Ansys Fluent uses a slight modification of Equation 12.101 (p. 472), derived by starting from Equation 12.93 (p. 470) and assuming that the droplet temperature is constant. This yields:

$$\frac{dm_p}{dt}h_{fg} = -hA_p\left(T_\infty - T_p\right) - A_p\varepsilon_p\sigma\left(\theta_R^{\ 4} - T_p^{\ 4}\right) \tag{12.102}$$

or

$$\frac{d\left(d_p\right)}{dt} = -\frac{2}{\rho_p h_{fg}}\left[\frac{k_\infty Nu}{d_p}\left(T_\infty - T_p\right) + \varepsilon_p\sigma\left(\theta_R^4 - T_p^4\right)\right] \tag{12.103}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Using Equation 12.76 (p. 465) for the Nusselt number correlation and replacing the Prandtl number term $0.3\mathrm{Pr}^{1/3}$ with the empirical constant 0.23, Equation 12.103 (p. 472) becomes

$$\frac{d\left(d_p\right)}{dt}=-\frac{2}{\rho_p h_{fg}}\left[\frac{2k_\infty\left[1+0.23\sqrt{Re_d}\right]}{d_p}\left(T_\infty-T_p\right)+\varepsilon_p\sigma\left(\theta_R^4-T_p^4\right)\right] \tag{12.104}$$

In the absence of radiation, this result matches that of Equation 12.101 (p. 472) in the limit that the argument of the logarithm is close to unity. Ansys Fluent uses Equation 12.104 (p. 473) when radiation is active in your model and Equation 12.101 (p. 472) when radiation is not active. Radiation heat transfer to the particle is included only if you have enabled the P-1 or discrete ordinates radiation model and you have activated radiation heat transfer to particles using the **Particle Radiation Interaction** option in the Discrete Phase Model Dialog Box.

The droplet is assumed to stay at constant temperature while the boiling rate is applied. Once the boiling law is entered it is applied for the duration of the particle trajectory. The energy required for vaporization appears as a (negative) source term in the energy equation for the gas phase. The evaporated liquid enters the gas phase as species i, as defined by your input for the destination species (see Setting Material Properties for the Discrete Phase in the User's Guide).

12.4.4. Devolatilization (Law 4)

The devolatilization law is applied to a combusting particle when the temperature of the particle reaches the vaporization temperature, T_{vap}, and remains in effect while the mass of the particle, m_p, exceeds the mass of the nonvolatiles in the particle:

$$T_p \geq T_{vap} \text{ and } T_p \geq T_{bp} \tag{12.105}$$

and

$$m_p > \left(1-f_{v,0}\right)\left(1-f_{w,0}\right)m_{p,0} \tag{12.106}$$

where $f_{w,0}$ is the mass fraction of the evaporating/boiling material if **Wet Combustion** is selected (otherwise, $f_{w,0}=0$). As implied by Equation 12.105 (p. 473), the boiling point, T_{bp}, and the vaporization temperature, T_{vap}, should be set equal to each other when Law 4 is to be used. When wet combustion is active, T_{bp} and T_{vap} refer to the boiling and evaporation temperatures for the droplet material only.

Ansys Fluent provides a choice of four devolatilization models:

- the constant rate model (the default model)
- the single kinetic rate model
- the two competing rates model (the Kobayashi model)
- the chemical percolation devolatilization (CPD) model

Each of these models is described, in turn, in the following sections.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

12.4.4.1. Choosing the Devolatilization Model

You will choose the devolatilization model when you are setting physical properties for the combusting-particle material in the Create/Edit Materials Dialog Box, as described in Description of the Properties in the User's Guide. By default, the constant rate model (Equation 12.107 (p. 474)) will be used.

12.4.4.2. The Constant Rate Devolatilization Model

The constant rate devolatilization law dictates that volatiles are released at a constant rate [50] (p. 1055):

$$-\frac{1}{f_{v,0}\left(1-f_{w,0}\right)m_{p,0}}\frac{dm_p}{dt}=A_0 \tag{12.107}$$

where

m_p = particle mass (kg)

$f_{v,0}$ = fraction of volatiles initially present in the particle

$m_{p,0}$ = initial particle mass (kg)

A_0 = rate constant (s^{-1})

The rate constant A_0 is defined in the material properties of the combusting particle(s). The Fluent materials database includes default values for each of the available combusting particle materials. A representative value is 12 s^{-1} derived from the work of Pillai [516] (p. 1083) on coal combustion. Proper use of the constant devolatilization rate requires that the vaporization temperature, which controls the onset of devolatilization, be set appropriately. Values in the literature show this temperature to be about 600 K [50] (p. 1055).

The volatile fraction of the particle enters the gas phase as the devolatilizing species i, defined by you (see Setting Material Properties for the Discrete Phase in the User's Guide). Once in the gas phase, the volatiles may react according to the inputs governing the gas phase chemistry.

12.4.4.3. The Single Kinetic Rate Model

The single kinetic rate devolatilization model assumes that the rate of devolatilization is first-order dependent on the amount of volatiles remaining in the particle [36] (p. 1055):

$$\frac{dm_p}{dt}=k\left[m_p-\left(1-f_{v,0}\right)\left(1-f_{w,0}\right)m_{p,0}\right] \tag{12.108}$$

where

m_p = particle mass (kg)

$f_{v,0}$ = mass fraction of volatiles initially present in the particle

$f_{w,0}$ = mass fraction of evaporating/boiling material (if wet combustion is modeled)

$m_{p,0}$ = initial particle mass (kg)

k = kinetic rate (s^{-1})

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Note that $f_{v,0}$, the fraction of volatiles in the particle, should be defined using a value slightly in excess of that determined by proximate analysis. The kinetic rate, k, is defined by input of an Arrhenius type pre-exponential factor and an activation energy:

$$k = A_1 e^{-(E/RT)} \tag{12.109}$$

The Fluent materials database includes default values for the rate constants, A_1 and E, for each of the available combusting particle materials.

Equation 12.108 (p. 474) has the approximate analytical solution:

$$\begin{aligned} m_p(t+\Delta t) &= \left(1-f_{v,0}\right)\left(1-f_{w,0}\right)m_{p,0} \\ &+\left[m_p(t)-\left(1-f_{v,0}\right)\left(1-f_{w,0}\right)m_{p,0}\right]e^{-k\Delta t} \end{aligned} \tag{12.110}$$

which is obtained by assuming that the particle temperature varies only slightly between discrete time integration steps.

Ansys Fluent can also solve Equation 12.108 (p. 474) in conjunction with the equivalent heat transfer equation using a stiff coupled solver. See Including Coupled Heat-Mass Solution Effects on the Particles in the User's Guide for details.

12.4.4.4. The Two Competing Rates (Kobayashi) Model

Ansys Fluent also provides the kinetic devolatilization rate expressions of the form proposed by Kobayashi [314] (p. 1071):

$$\mathfrak{R}_1 = A_1 e^{-(E_1/RT_p)} \tag{12.111}$$

$$\mathfrak{R}_2 = A_2 e^{-(E_2/RT_p)} \tag{12.112}$$

where $\mathfrak{R}_1$ and $\mathfrak{R}_2$ are competing rates that may control the devolatilization over different temperature ranges. The two kinetic rates are weighted to yield an expression for the devolatilization as

$$\frac{m_v(t)}{\left(1-f_{w,0}\right)m_{p,0}-m_a} = \int_0^t \left(\alpha_1\mathfrak{R}_1+\alpha_2\mathfrak{R}_2\right)\exp\left(-\int_0^t\left(\mathfrak{R}_1+\mathfrak{R}_2\right)dt\right)dt \tag{12.113}$$

where

$m_v(t)$ = volatile yield up to time t

$m_{p,0}$ = initial particle mass at injection

α_1, α_2 = yield factors

m_a = ash content in the particle

The Kobayashi model requires input of the kinetic rate parameters, A_1, E_1, A_2, and E_2, and the yields of the two competing reactions, α_1 and α_2. Ansys Fluent uses default values for the yield factors of 0.3 for the first (slow) reaction and 1.0 for the second (fast) reaction. It is recommended in the literature [314] (p. 1071) that α_1 be set to the fraction of volatiles determined by proximate analysis, since this rate represents devolatilization at low temperature. The second yield parameter, α_2, should be set close to unity, which is the yield of volatiles at very high temperature.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Note that in Ansys Fluent, the effect of particle heating rate on evolution of volatile yield is not considered.

By default, Equation 12.113 (p. 475) is integrated in time analytically, assuming the particle temperature to be constant over the discrete time integration step. Ansys Fluent can also solve Equation 12.113 (p. 475) in conjunction with the equivalent heat transfer equation using a stiff coupled solver. See Including Coupled Heat-Mass Solution Effects on the Particles in the User's Guide for details.

12.4.4.5. The CPD Model

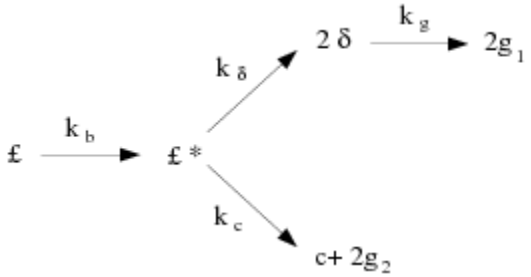
In contrast to the coal devolatilization models presented above, which are based on empirical rate relationships, the chemical percolation devolatilization (CPD) model characterizes the devolatilization behavior of rapidly heated coal based on the physical and chemical transformations of the coal structure [183] (p. 1063), [184] (p. 1063), [218] (p. 1065).

12.4.4.5.1. General Description

During coal pyrolysis, the labile bonds between the aromatic clusters in the coal structure lattice are cleaved, resulting in two general classes of fragments. One set of fragments has a low molecular weight (and correspondingly high vapor pressure) and escapes from the coal particle as a light gas. The other set of fragments consists of tar gas precursors that have a relatively high molecular weight (and correspondingly low vapor pressure) and tend to remain in the coal for a long period of time during typical devolatilization conditions. During this time, reattachment with the coal lattice (which is referred to as cross-linking) can occur. The high molecular weight compounds plus the residual lattice are referred to as metaplast. The softening behavior of a coal particle is determined by the quantity and nature of the metaplast generated during devolatilization. The portion of the lattice structure that remains after devolatilization is are made up of char and mineral-compound-based ash.

The CPD model characterizes the chemical and physical processes by considering the coal structure as a simplified lattice or network of chemical bridges that link the aromatic clusters. Modeling the cleavage of the bridges and the generation of light gas, char, and tar precursors is then considered to be analogous to the chemical reaction scheme shown in Figure 12.1: Coal Bridge (p. 476).

Figure 12.1: Coal Bridge



The variable $£$ represents the original population of labile bridges in the coal lattice. Upon heating, these bridges become the set of reactive bridges, $£^*$. For the reactive bridges, two competing paths are available. In one path, the bridges react to form side chains, δ. The side chains may detach from the aromatic clusters to form light gas, g_1. As bridges between neighboring aromatic clusters are cleaved, a certain fraction of the coal becomes detached from the coal lattice.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

These detached aromatic clusters are the heavy-molecular-weight tar precursors that form the metaplast. The metaplast vaporizes to form coal tar. While waiting for vaporization, the metaplast can also reattach to the coal lattice matrix (crosslinking). In the other path, the bridges react and become a char bridge, c, with the release of an associated light gas product, g_2. The total population of bridges in the coal lattice matrix can be represented by the variable p, where $p=£+c$.

12.4.4.5.2. Reaction Rates

Given this set of variables that characterizes the coal lattice structure during devolatilization, the following set of reaction rate expressions can be defined for each, starting with the assumption that the reactive bridges are destroyed at the same rate at which they are created ($\frac{\partial £^*}{\partial t}=0$):

$$\frac{d£}{dt}=-k_b£ \tag{12.114}$$

$$\frac{dc}{dt}=k_b\frac{£}{\rho+1} \tag{12.115}$$

$$\frac{d\delta}{dt}=\left[2\rho k_b\frac{£}{\rho+1}\right]-k_g\delta \tag{12.116}$$

$$\frac{dg_1}{dt}=k_g\delta \tag{12.117}$$

$$\frac{dg_2}{dt}=2\frac{dc}{dt} \tag{12.118}$$

where the rate constants for bridge breaking and gas release steps, k_b and k_g, are expressed in Arrhenius form with a distributed activation energy:

$$k=Ae^{-\left(E\pm E_\sigma\right)/RT} \tag{12.119}$$

where A, E, and E_σ are, respectively, the pre-exponential factor, the activation energy, and the distributed variation in the activation energy, R is the universal gas constant, and T is the temperature. The ratio of rate constants, $\rho=k_\delta/k_c$, is set to 0.9 in this model based on experimental data.

12.4.4.5.3. Mass Conservation

The following mass conservation relationships are imposed:

$$g = g_1+g_2 \tag{12.120}$$

$$g_1 = 2f-\sigma \tag{12.121}$$

$$g_2 = 2\left(c-c_0\right) \tag{12.122}$$

where f is the fraction of broken bridges ($f=1-p$). The initial conditions for this system are given by the following:

$$c\left(0\right) = c_0 \tag{12.123}$$

$$£\left(0\right) = £_0=p_0-c_0 \tag{12.124}$$

$$\delta\left(0\right) = 2f_0=2\left(1-c_0-£_0\right) \tag{12.125}$$

$$g\left(0\right) = g_1\left(0\right)=g_2\left(0\right)=0 \tag{12.126}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

where c_0 is the initial fraction of char bridges, p_0 is the initial fraction of bridges in the coal lattice, and $£_0$ is the initial fraction of labile bridges in the coal lattice.

12.4.4.5.4. Fractional Change in the Coal Mass

Given the set of reaction equations for the coal structure parameters, it is necessary to relate these quantities to changes in coal mass and the related release of volatile products. To accomplish this, the fractional change in the coal mass as a function of time is divided into three parts: light gas (f_{gas}), tar precursor fragments (f_{frag}), and char (f_{char}). This is accomplished by using the following relationships, which are obtained using percolation lattice statistics:

$$f_{gas}(t)=\frac{r\left(g_1+g_2\right)(\sigma+1)}{4+2r(1-c_0)(\sigma+1)} \tag{12.127}$$

$$f_{frag}(t)=\frac{2}{2+r(1-c_0)(\sigma+1)}\left[\Phi F(p)+r\Omega K(p)\right] \tag{12.128}$$

$$f_{char}(t)=1-f_{gas}(t)-f_{frag}(t) \tag{12.129}$$

The variables Φ, Ω, $F(p)$, and $K(p)$ are the statistical relationships related to the cleaving of bridges based on the percolation lattice statistics, and are given by the following equations:

$$\Phi=1+r\left[\frac{£}{p}+\frac{(\sigma-1)\delta}{4(1-p)}\right] \tag{12.130}$$

$$\Omega=\frac{\delta}{2(1-p)}-\frac{£}{p} \tag{12.131}$$

$$F(p)=\left(\frac{p'}{p}\right)^{\frac{\sigma+1}{\sigma-1}} \tag{12.132}$$

$$K(p)=\left[1-\left(\frac{\sigma+1}{2}\right)p'\right]\left(\frac{p'}{p}\right)^{\frac{\sigma+1}{\sigma-1}} \tag{12.133}$$

r is the ratio of bridge mass to site mass, $m_b/\ m_a$, where

$$m_b \ = \ 2M_{w,\delta} \tag{12.134}$$

$$m_a \ = \ M_{w,1}-(\sigma+1)M_{w,\delta} \tag{12.135}$$

where $M_{w,\delta}$ and $M_{w,1}$ are the side chain and cluster molecular weights respectively. $\sigma+1$ is the lattice coordination number, which is determined from solid-state nuclear magnetic Resonance (NMR) measurements related to coal structure parameters, and p' is the root of the following equation in p (the total number of bridges in the coal lattice matrix):

$$p'(1-p')^{\sigma-1}=p(1-p)^{\sigma-1} \tag{12.136}$$

In accounting for mass in the metaplast (tar precursor fragments), the part that vaporizes is treated in a manner similar to flash vaporization, where it is assumed that the finite fragments undergo vapor/liquid phase equilibration on a time scale that is rapid with respect to the bridge reactions. As an estimate of the vapor/liquid that is present at any time, a vapor pressure correlation based on a simple form of Raoult's Law is used. The vapor pressure treatment is largely responsible for predicting pressure-dependent devolatilization yields. For the part of the metaplast that reattaches to the coal lattice, a cross-linking rate expression given by the following equation is used:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$\frac{dm_{cross}}{dt} = m_{frag} A_{cross} e^{-(E_{cross}/RT)} \tag{12.137}$$

where m_{cross} is the amount of mass reattaching to the matrix, m_{frag} is the amount of mass in the tar precursor fragments (metaplast), and A_{cross} and E_{cross} are rate expression constants.

12.4.4.5.5. CPD Inputs

Given the set of equations and corresponding rate constants introduced for the CPD model, the number of constants that must be defined to use the model is a primary concern. For the relationships defined previously, it can be shown that the following parameters are coal independent [183] (p. 1063):

- A_b, E_b, $E_{\sigma b}$, A_g, E_g, and $E_{\sigma g}$ for the rate constants k_b and k_g
- A_{cross}, E_{cross}, and ρ

These constants are included in the submodel formulation and are not input or modified during problem setup.

There are an additional five parameters that are coal-specific and must be specified during the problem setup:

- initial fraction of bridges in the coal lattice, p_0
- initial fraction of char bridges, c_0
- lattice coordination number, $\sigma+1$
- cluster molecular weight, $M_{w,1}$
- side chain molecular weight, $M_{w,\delta}$

The first four of these are coal structure quantities that are obtained from NMR experimental data. The last quantity, representing the char bridges that either exist in the parent coal or are formed very early in the devolatilization process, is estimated based on the coal rank. These quantities are entered in the **Create/Edit Materials** dialog box, as described in Description of the Properties in the User's Guide. Values for the coal-dependent parameters for a variety of coals are listed in Table 12.1: Chemical Structure Parameters for C NMR for 13 Coals (p. 479).

Table 12.1: Chemical Structure Parameters for C NMR for 13 Coals

Coal Type	$\sigma+1$	p_0	$M_{w,1}$	$M_{w,\delta}$	c_0
Zap (AR)	3.9	.63	277	40	.20
Wyodak (AR)	5.6	.55	410	42	.14
Utah (AR)	5.1	.49	359	36	0
Ill6 (AR)	5.0	.63	316	27	0
Pitt8 (AR)	4.5	.62	294	24	0
Stockton (AR)	4.8	.69	275	20	0
Freeport (AR)	5.3	.67	302	17	0

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Pocahontas (AR)	4.4	.74	299	14	.20
Blue (Sandia)	5.0	.42	410	47	.15
Rose (AFR)	5.8	.57	459	48	.10
1443 (lignite, ACERC)	4.8	.59	297	36	.20
1488 (subbituminous, ACERC)	4.7	.54	310	37	.15
1468 (anthracite, ACERC)	4.7	.89	656	12	.25

AR refers to eight types of coal from the Argonne premium sample bank [613] (p. 1088), [678] (p. 1092). Sandia refers to the coal examined at Sandia National Laboratories [182] (p. 1063). AFR refers to coal examined at Advanced Fuel Research. ACERC refers to three types of coal examined at the Advanced Combustion Engineering Research Center.

12.4.4.6. Particle Swelling During Devolatilization

The particle diameter changes during devolatilization according to the swelling coefficient, C_{sw}, which is defined by you and applied in the following relationship:

$$\frac{d_p}{d_{p,0}} = 1 + \left(C_{sw} - 1\right)\frac{\left(1 - f_{w,0}\right)m_{p,0} - m_p}{f_{v,0}\left(1 - f_{w,0}\right)m_{p,0}} \tag{12.138}$$

where

$d_{p,0}$ = particle diameter at the start of devolatilization

d_p = current particle diameter

The term $\left(1 - f_{w,0}\right)m_{p,0} - m_p \Big/ f_{v,0}\left(1 - f_{w,0}\right)m_{p,0}$ is the ratio of the mass that has been devolatilized to the total volatile mass of the particle. This quantity approaches a value of 1.0 as the devolatilization law is applied. When the swelling coefficient is equal to 1.0, the particle diameter stays constant. When the swelling coefficient is equal to 2.0, the final particle diameter doubles when all of the volatile component has vaporized, and when the swelling coefficient is equal to 0.5 the final particle diameter is half of its initial diameter.

12.4.4.7. Heat Transfer to the Particle During Devolatilization

Heat transfer to the particle during the devolatilization process includes contributions from convection and radiation (if active):

$$m_p c_p \frac{dT_p}{dt} = hA_p\left(T_\infty - T_p\right) + A_p \epsilon_p \sigma\left(\theta_R{}^4 - T_p{}^4\right) \tag{12.139}$$

where the variables have already been defined for Equation 12.93 (p. 470).

Radiation heat transfer to the particle is included only if you have enabled the P-1 or discrete ordinates radiation heat transfer to particles using the **Particle Radiation Interaction** option in the Discrete Phase Model Dialog Box.

By default, Equation 12.139 (p. 480) is solved analytically, by assuming that the temperature and mass of the particle do not change significantly between time steps:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$T_p(t+\Delta t)=\alpha_p+\left[T_p(t)-\alpha_p\right]e^{-\beta_p \Delta t} \tag{12.140}$$

where

$$\alpha_p=\frac{hA_pT_\infty+A_p\epsilon_p\sigma\theta_R{}^4}{hA_p+\epsilon_pA_p\sigma T_p{}^3} \tag{12.141}$$

and

$$\beta_p=\frac{A_p\left(h+\varepsilon_p\sigma T_p{}^3\right)}{m_pc_p} \tag{12.142}$$

Ansys Fluent can also solve Equation 12.139 (p. 480) in conjunction with the equivalent mass transfer equation using a stiff coupled solver. See Including Coupled Heat-Mass Solution Effects on the Particles in the User's Guide for details.

12.4.5. Surface Combustion (Law 5)

After the volatile component of the particle is completely evolved, a surface reaction begins that consumes the combustible fraction, f_{comb}, of the particle. Law 5 is therefore active (for a combusting particle) after the volatiles are evolved:

$$m_p<\left(1-f_{v,0}\right)\left(1-f_{w,0}\right)m_{p,0} \tag{12.143}$$

and until the combustible fraction is consumed:

$$m_p>\left(1-f_{v,0}-f_{comb}\right)\left(1-f_{w,0}\right)m_{p,0} \tag{12.144}$$

When the combustible fraction, f_{comb}, has been consumed in Law 5, the combusting particle may contain residual "ash" that reverts to the inert heating law, Law 6 (described previously).

With the exception of the multiple surface reactions model, the surface combustion law consumes the reactive content of the particle as governed by the stoichiometric requirement, S_b, of the surface "burnout" reaction:

$$char(s)+S_box(g)\rightarrow products(g) \tag{12.145}$$

where S_b is defined in terms of mass of oxidant per mass of char, and the oxidant and product species are defined in the Set Injection Properties Dialog Box.

Ansys Fluent provides a choice of four heterogeneous surface reaction rate models for combusting particles:

- the diffusion-limited rate model (the default model)
- the kinetics/diffusion-limited rate model
- the intrinsic model
- the multiple surface reactions model

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Each of these models is described in detail below. You will choose the surface combustion model when you are setting physical properties for the combusting-particle material, as described in Description of the Properties in the User's Guide. By default, the diffusion-limited rate model will be used.

12.4.5.1. The Diffusion-Limited Surface Reaction Rate Model

The diffusion-limited surface reaction rate model, which is the default model in Ansys Fluent, assumes that the surface reaction proceeds at a rate determined by the diffusion of the gaseous oxidant to the surface of the particle:

$$\frac{dm_p}{dt} = -4\pi d_p D_{i,m} \frac{Y_{ox} T_\infty \rho}{S_b \left(T_p + T_\infty \right)} \tag{12.146}$$

where

$D_{i,m}$ = diffusion coefficient for oxidant in the bulk (m^2/s)

Y_{ox} = local mass fraction of oxidant in the gas

ρ = gas density (kg/m^3)

S_b = stoichiometry of Equation 12.145 (p. 481)

Equation 12.146 (p. 482) is derived from the model of Baum and Street [50] (p. 1055) with the kinetic contribution to the surface reaction rate ignored. The diffusion-limited rate model assumes that the diameter of the particles does not change. Since the mass of the particles is decreasing, the effective density decreases, and the char particles become more porous.

12.4.5.2. The Kinetic/Diffusion Surface Reaction Rate Model

The kinetic/diffusion-limited rate model assumes that the surface reaction rate is determined either by kinetics or by a diffusion rate. Ansys Fluent uses the model of Baum and Street [50] (p. 1055) and Field [179] (p. 1063), in which a diffusion rate coefficient

$$D_0 = C_1 \frac{\left[\left(T_p + T_\infty \right) / 2 \right]^{0.75}}{d_p} \tag{12.147}$$

and a kinetic rate

$$\mathcal{R} = C_2 e^{-\left(E / RT_p \right)} \tag{12.148}$$

are weighted to yield a char combustion rate of

$$\frac{dm_p}{dt} = -A_p p_{ox} \frac{D_0 \mathcal{R}}{D_0 + \mathcal{R}} \tag{12.149}$$

where A_p is the surface area of the droplet (πd_p^2), p_{ox} is the partial pressure of oxidant species in the gas surrounding the combusting particle, and the kinetic rate, $\mathcal{R}$, incorporates the effects of chemical reaction on the internal surface of the char particle (intrinsic reaction) and pore diffusion. In Ansys Fluent, Equation 12.149 (p. 482) is recast in terms of the oxidant mass fraction, Y_{ox}, as

$$\frac{dm_p}{dt} = -A_p \frac{\rho R T_\infty Y_{ox}}{M_{w,ox}} \frac{D_0 \mathcal{R}}{D_0 + \mathcal{R}} \tag{12.150}$$

The particle size is assumed to remain constant in this model while the density is allowed to decrease.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

When this model is enabled, the rate constants used in Equation 12.147 (p. 482) and Equation 12.148 (p. 482) are entered in the Create/Edit Materials Dialog Box, as described in Setting Material Properties for the Discrete Phase in the User's Guide.

12.4.5.3. The Intrinsic Model

The intrinsic model in Ansys Fluent is based on Smith's model [606] (p. 1088), assuming the order of reaction is equal to unity. Like the kinetic/diffusion model, the intrinsic model assumes that the surface reaction rate includes the effects of both bulk diffusion and chemical reaction (see Equation 12.150 (p. 482)). The intrinsic model uses Equation 12.147 (p. 482) to compute the diffusion rate coefficient, D_0, but the chemical rate, $\mathfrak{R}$, is explicitly expressed in terms of the intrinsic chemical and pore diffusion rates:

$$\mathfrak{R}=\eta\frac{d_p}{6}\rho_p A_g k_i \tag{12.151}$$

η is the effectiveness factor, or the ratio of the actual combustion rate to the rate attainable if no pore diffusion resistance existed [342] (p. 1073):

$$\eta=\frac{3}{\varphi^2}\left(\varphi\coth\varphi-1\right) \tag{12.152}$$

where φ is the Thiele modulus:

$$\varphi=\frac{d_p}{2}\left[\frac{S_b\rho_p A_g k_i p_{ox}}{D_e\rho_{ox}}\right]^{1/2} \tag{12.153}$$

ρ_{ox} is the density of oxidant in the bulk gas (kg/m^3) and D_e is the effective diffusion coefficient in the particle pores. Assuming that the pore size distribution is unimodal and the bulk and Knudsen diffusion proceed in parallel, D_e is given by

$$D_e=\frac{\theta}{\tau^2}\left[\frac{1}{D_{Kn}}+\frac{1}{D_0}\right]^{-1} \tag{12.154}$$

where D_0 is the bulk molecular diffusion coefficient and θ is the porosity of the char particle:

$$\theta=1-\frac{\rho_p}{\rho_t} \tag{12.155}$$

ρ_p and ρ_t are, respectively, the apparent and true densities of the pyrolysis char.

τ (in Equation 12.154 (p. 483)) is the tortuosity of the pores. The default value for τ in Ansys Fluent is $\sqrt{2}$, which corresponds to an average intersecting angle between the pores and the external surface of 45° [342] (p. 1073).

D_{Kn} is the Knudsen diffusion coefficient:

$$D_{Kn}=97.0\bar{r}_p\sqrt{\frac{T_p}{M_{w,ox}}} \tag{12.156}$$

where T_p is the particle temperature and $\bar{r}_p$ is the mean pore radius of the char particle, which can be measured by mercury porosimetry. Note that macropores ($\bar{r}_p>150$ Å) dominate in low-rank chars while micropores ($\bar{r}_p<10$ Å) dominate in high-rank chars [342] (p. 1073).

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

A_g (in Equation 12.151 (p. 483) and Equation 12.153 (p. 483)) is the specific internal surface area of the char particle, which is assumed in this model to remain constant during char combustion. Internal surface area data for various pyrolysis chars can be found in [605] (p. 1088). The mean value of the internal surface area during char combustion is higher than that of the pyrolysis char [342] (p. 1073). For example, an estimated mean value for bituminous chars is 300 m^2/g [101] (p. 1058).

k_i (in Equation 12.151 (p. 483) and Equation 12.153 (p. 483)) is the intrinsic reactivity, which is of Arrhenius form:

$$k_i = A_i e^{-\left(E_i / RT_p\right)} \tag{12.157}$$

where the pre-exponential factor A_i and the activation energy E_i can be measured for each char. In the absence of such measurements, the default values provided by Ansys Fluent (which are taken from a least squares fit of data of a wide range of porous carbons, including chars [605] (p. 1088)) can be used.

To allow a more adequate description of the char particle size (and hence density) variation during combustion, you can specify the burning mode α, relating the char particle diameter to the fractional degree of burnout U (where $U = 1 - m_p / m_{p,0}$) by [604] (p. 1088)

$$\frac{d_p}{d_{p,0}} = \left(1 - U\right)^{\alpha} \tag{12.158}$$

where m_p is the char particle mass and the subscript zero refers to initial conditions (that is, at the start of char combustion). Note that $0 \le \alpha \le 1/3$ where the limiting values 0 and $1/3$ correspond, respectively, to a constant size with decreasing density (zone 1) and a decreasing size with constant density (zone 3) during burnout. In zone 2, an intermediate value of $\alpha = 0.25$, corresponding to a decrease of both size and density, has been found to work well for a variety of chars [604] (p. 1088).

When this model is enabled, the rate constants used in Equation 12.147 (p. 482), Equation 12.151 (p. 483), Equation 12.153 (p. 483), Equation 12.154 (p. 483), Equation 12.156 (p. 483), Equation 12.157 (p. 484), and Equation 12.158 (p. 484) are entered in the Create/Edit Materials Dialog Box, as described in Setting Material Properties for the Discrete Phase in the User's Guide.

12.4.5.4. The Multiple Surface Reactions Model

Modeling multiple particle surface reactions follows a pattern similar to the wall surface reaction models, where the surface species is now a "particle surface species". For the mixture material defined in the **Species Model** dialog box, the particle surface species can be depleted or produced by the stoichiometry of the particle surface reaction (defined in the **Reactions** dialog box). The particle surface species constitutes the reactive char mass of the particle, hence, if a particle surface species is depleted, the reactive "char" content of the particle is consumed, and in turn, when a surface species is produced, it is added to the particle "char" mass. Any number of particle surface species and any number of particle surface reactions can be defined for any given combusting particle.

Multiple injections can be accommodated, and combusting particles reacting according to the multiple surface reactions model can coexist in the calculation, with combusting particles following other char combustion laws. The model is based on oxidation studies of char particles, but it is also applicable to gas-solid reactions in general, not only to char oxidation reactions.

See Combusting Particle Surface Reactions (p. 261) for information about particle surface reactions.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

12.4.5.4.1. Limitations

Note the following limitations of the multiple surface reactions model:

- The model is available only with the species transport model for volumetric reactions, and not with the nonpremixed, premixed, or partially premixed combustion models.

12.4.5.5. Heat and Mass Transfer During Char Combustion

The surface reaction consumes the oxidant species in the gas phase; that is, it supplies a (negative) source term during the computation of the transport equation for this species. Similarly, the surface reaction is a source of species in the gas phase: the product of the heterogeneous surface reaction appears in the gas phase as a user-selected chemical species. The surface reaction also consumes or produces energy, in an amount determined by the heat of reaction defined by you.

The particle heat balance during surface reaction is

$$m_p c_p \frac{dT_p}{dt} = hA_p\left(T_\infty - T_p\right) - f_h \frac{dm_p}{dt} H_{reac} + A_p \varepsilon_p \sigma\left(\theta_R^4 - T_p^4\right) \tag{12.159}$$

where H_{reac} is the heat released by the surface reaction. Note that only a portion $(1-f_h)$ of the energy produced by the surface reaction appears as a heat source in the gas-phase energy equation: the particle absorbs a fraction f_h of this heat directly. For coal combustion, it is recommended that f_h be set to 1.0 if the char burnout product is CO and 0.3 if the char burnout product is CO_2 [73] (p. 1057).

For heat exchange coefficient correlations available in Ansys Fluent, refer to Inert Heating or Cooling (Law 1/Law 6) (p. 463).

Radiation heat transfer to the particle is included only if you have enabled the P-1 or discrete ordinates radiation heat transfer to particles using the **Particle Radiation Interaction** option in the Discrete Phase Model Dialog Box.

By default, Equation 12.159 (p. 485) is solved analytically, by assuming that the temperature and mass of the particle do not change significantly between time steps. Ansys Fluent can also solve Equation 12.159 (p. 485) in conjunction with the equivalent mass transfer equation using a stiff coupled solver. See Including Coupled Heat-Mass Solution Effects on the Particles in the User's Guide for details.

12.4.6. Multicomponent Particle Definition (Law 7)

Multicomponent particles are described in Ansys Fluent as droplet particles containing a mixture of several components or species. The particle mass m is the sum of the masses of the components

$$m = \sum_i m_i \tag{12.160}$$

The density of the particle ρ_p can be either constant, or volume-averaged:

$$\rho_p = \left(\sum_i \frac{m_i}{m\rho_i}\right)^{-1} \tag{12.161}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

The multicomponent droplet vaporization rate is calculated as the sum of the vaporization rates of the individual components.

For the **diffusion-controlled** vaporization model, the vaporization rate of component i is given by Equation 12.162 (p. 486)

$$\frac{dm_i}{dt} = -A_p M_{w,i} k_{c,i} \left(C_{i,s} - C_{i,\infty} \right) \tag{12.162}$$

while for the **convection/diffusion controlled** model, it is given by:

$$\frac{dm_i}{dt} = -A_p k_{c,i} \rho \ln\left(1 + B_{m,i}\right) \tag{12.163}$$

where

m_i = mass of component i in droplet (kg)

$k_{c,i}$ = mass transfer coefficient of component i (m/s) given by Equation 12.83 (p. 466)

A_p = droplet surface area (m^2)

ρ = gas density (kg/m^3)

$M_{w,i}$ = molecular weight of species i

$C_{i,s}$ and $C_{i,\infty}$ = concentration of species i at the droplet surface and the bulk, respectively ($kmol/m^3$).

$B_{m,i}$ = Spalding mass number for species i and is given by Equation 12.86 (p. 467)

For the single-rate thermolysis model [67] (p. 1056), the vaporization rate of component i is given by:

$$\frac{dm_i}{dt} = -\pi d_p A_i e^{-E_i/RT_p} \tag{12.164}$$

where

m_i = mass of component i (kg)

d_p = droplet diameter (m)

A_i = pre-exponential factor (kg/s-m)

E_i = activation energy (J/kg)

T_p = particle temperature (K)

A simplified constant rate thermolysis model is also available:

$$\frac{dm_p}{dt} = -C_i m_{p0} \tag{12.165}$$

where

C_i = rate constant (1/s)

m_{p0} = initial mass of component i (kg)

Note that the Thermolysis model rate applies to the individual droplet component of the multicomponent particle.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

The equation for the multicomponent particle temperature T consists of terms for radiation, convective heating and vaporization, and is cast similarly to the single component particle energy equation (Equation 12.70 (p. 464)). Radiation heat transfer to the particle is included only if you have enabled **P-1** or **Discrete-Ordinates (DO)** radiation heat transfer to the particles using the **Particle Radiation Interaction** option in the **Discrete Phase Model** dialog box.

The energy equation is written for the multicomponent particle as follows:

$$m_p c_p \frac{dT}{dt} = A_p \varepsilon_p \sigma \left(\Theta_R^4 - T_p^4\right) + h A_p \left(T_\infty - T_p\right) + \sum_i \frac{dm_i}{dt} \left(h_{vap,i}\right) \tag{12.166}$$

where

dm_i/dt is computed from Equation 12.162 (p. 486), Equation 12.163 (p. 486), or Equation 12.168 (p. 487)

$h_{vap,i}$ = latent heat of vaporization for component i

h = heat transfer coefficient computed from:

- Equation 12.76 (p. 465) for the **diffusion-controlled** vaporization model
- Equation 12.94 (p. 470) for the **convection/diffusion-controlled** vaporization model and the boiling regime

When radiation heat transfer is active, the boiling rate dm_i/dt calculated by Equation 12.168 (p. 487), is augmented by the radiation term,

$$\frac{x_i}{h_{vap,i}} A_p \varepsilon_p \sigma \left(\Theta_R^4 - T_p^4\right) \tag{12.167}$$

When the total vapor pressure at the multicomponent droplet surface exceeds the cell pressure, Ansys Fluent applies a boiling rate equation ([344] (p. 1073), [495] (p. 1082), [13] (p. 1053)). The total vapor pressure is computed as $P_t = \sum P_i$ where P_i is the partial pressure of component i.

An equation for the boiling rate $\dot{m}_i$ for component i can be formulated from the overall energy balance by setting $dT/dt = 0$ following the assumption that the particle temperature is changing very slowly in the boiling regime:

$$\dot{m}_i = \varepsilon_i \frac{\pi d_p k}{c_p} \left(2 + 0.6 Re_p^{\frac{1}{2}} Pr^{\frac{1}{3}}\right) \ln\left(1 + B_{T,ave}\right) \tag{12.168}$$

Where

$$B_{T,ave} = \frac{c_p \left(T_\infty - T_p\right)}{h_{vap,ave}} \tag{12.169}$$

$$h_{vap,ave} = \sum \varepsilon_i h_{vap,i} \tag{12.170}$$

The fractional vaporization rate ε_i is computed as the normalized mass fraction of the species i at the particle surface:

$$\varepsilon_i = \frac{y_{surf,i}}{\sum_i y_{surf,i}} \tag{12.171}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

where $y_{surf,i}$ is known from the vapor-liquid equilibrium (VLE) expression.

In the above equations, the following notation is used:

d_p = droplet diameter (m)

k = gas thermal conductivity (W/m–K)

c_p = gas specific heat (J/kg–K)

12.4.6.1. Raoult's Law

The correlation between the vapor concentration of a species $C_{i,s}$ over the surface and its mole fraction in the condensed phase x_i^L is described by Raoult's law:

$$C_{i,s}=\frac{p_i}{RT}=\frac{x_i^L p_{sat,i}}{RT} \tag{12.172}$$

Raoult's law assumes that the vapor phase is an ideal gas and the liquid particle mixture is an ideal mixture. Liquid mixtures close to ideal are those of similar chemical species (for example, n-pentane/n-hexane, or benzene/toluene). Raoult's law may also be valid when x_i^L approaches 1 for the solvent species i when a non-volatile component (small mass fraction) is dissolved in a volatile solvent (large mass fraction). Refer to Vapor Liquid Equilibrium Theory (p. 488) for more details on the assumptions and the derivation of Raoult's law. Note that Equation 12.172 (p. 488) is not applied for the components that have been defined as immiscible-not-vaporizing.

12.4.6.2. Peng-Robinson Real Gas Model

In the Peng-Robinson Real Gas Model, the vapor concentration of each species at the surface is deduced from the calculations of vapor mole fraction and compressibility:

$$C_{i,S}=x_i{}^V\frac{p}{Z^V RT} \tag{12.173}$$

These calculations are described in Vapor Liquid Equilibrium Theory (p. 488). The vapor-liquid equilibrium is deduced from the inputs for the Peng-Robinson model in the evaporating vapor species, so it is not necessary to specify vapor pressure separately.

Note that Equation 12.173 (p. 488) is not applied for the components that have been defined as immiscible-not-vaporizing.

Instead of using Raoult's Law or the Peng-Robinson equation of state, you can define your own user-defined function for delivering the vapor concentration at the particle surface. For more information, see `DEFINE_DPM_VP_EQUILIB` in the Fluent Customization Manual.

12.5. Vapor Liquid Equilibrium Theory

A number of industrially important processes, such as distillation, absorption, and extraction, bring into contact two phases that are not at equilibrium. The rate at which a species is transferred from one phase to the other depends on the departure of the system from equilibrium. The quantitative treatment of these rate processes requires knowledge of the equilibrium states of the system. Apart from these cases, vapor-liquid equilibrium (VLE) relationships in multicomponent systems are needed for the solution

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

of many other classes of engineering problems, such as the computation of evaporation rates in spray combustion applications.

For the multicomponent particle, the vaporization rate of component i is computed from Equation 12.162 (p. 486) or Equation 12.163 (p. 486) and requires knowledge of the component concentration or the mass fraction at the droplet surface.

The concentration or mass fraction of species i at the particle surface is computed from the mole fraction x_i, which can be obtained from the general expression for two phase equilibrium, equating the fugacity of the liquid and vapor mixture components [522] (p. 1083):

$$f_i^V = x_i^V \varphi_i^V p = x_i^L \varphi_i^L p = f_i^L \tag{12.174}$$

where x_i is the mole fraction, φ_i is the fugacity coefficient for the species i in the mixture, and p is the absolute pressure. The superscripts V and L are the vapor and the liquid phase variables, respectively. The fugacity coefficients account for the nonideality in the gas and liquid mixture. The fugacity of the liquid phase can be calculated from the pure component's saturation pressure $p_{sat,i}$ [607] (p. 1088):

$$f_i^L = x_i^L \varphi_i^L p = \gamma_i x_i^L \varphi_{sat,i}^L p_{sat,i} exp\left[\frac{V_i^L\left(p - p_{sat,i}\right)}{RT}\right] \tag{12.175}$$

Here, $\varphi_{sat,i}^L$ is the fugacity coefficient for pure i at the saturation pressure; γ_i is the activity coefficient for species i in the mixture, and accounts for the nonideality in the liquid phase; T is the particle surface temperature. We assume perfect thermal conductivity inside the particle, so the particle temperature is used instead; R is the universal gas constant; V_i^L is the molar volume of the liquid. The exponential term is the Poynting correction factor and accounts for the compressibility effects within the liquid. Except at high pressures, the Poynting factor is usually negligible. Under low pressure conditions where the gas phase may be assumed to be ideal, $\varphi_i^V \approx 1$ and $\varphi_{sat,i} \approx 1$. Furthermore, if the liquid is also assumed to be ideal, $\gamma \approx 1$, then Equation 12.174 (p. 489) reduces to Raoult's law,

$$x_i^V p = x_i^L p_{sat,i} \tag{12.176}$$

Raoult's law is the default vapor-liquid equilibrium expression used in the Ansys Fluent multicomponent droplet model. However, there is a UDF hook available for user-defined vapor-liquid equilibrium models.

While Raoult's law represents the simplest form of the VLE equation, keep in mind that it is of limited use, as the assumptions made for its derivation are usually unrealistic. The most critical assumption is that the liquid phase is an ideal solution. This is not likely to be valid, unless the system is made up of species of similar molecular sizes and chemical nature, such as in the case of benzene and toluene, or n-heptane and n-hexane.

For higher pressures, especially near or above the critical point of the components, real gas effects must be considered. Most models describing the fugacity coefficients use a cubic equation of state with the general form:

$$p = \frac{RT}{V-b} - \frac{a\left(V-\eta\right)}{\left(V-b\right)\left(V^2+\delta V+\varepsilon\right)} \tag{12.177}$$

where V is the molar volume. For in-cylinder applications, the Peng-Robinson equation of state is often used [507] (p. 1082), where $\delta = 2b$, $\varepsilon = -b^2$, and $\eta = b$:

$$p=\frac{RT}{V-b}-\frac{a}{V^2+2bV-b^2} \tag{12.178}$$

This equation defines the compressibility

$$Z=\frac{pV}{RT}=\frac{V}{V-b}-\frac{aV/RT}{V^2+2bV-b^2} \tag{12.179}$$

The implementation of the Peng-Robinson equation of state in Ansys Fluent uses this expression for both phases, the particle liquid and the vapor phase. The parameters a and b are determined by the composition using a simple mixing law:

$$\begin{aligned} a&=\sum_{i=1}^{N}\sum_{j=1}^{N}x_i x_j\sqrt{a_i a_j} \\ b&=\sum_{i=1}^{N}x_i b_i \end{aligned} \tag{12.180}$$

where N is the number of components in the mixture. The pure component parameters can be obtained using the relationship with the Peng-Robinson constants:

$$\begin{aligned} a_i&=\left(0.457247\frac{R^2T_{c,i}^2}{p_{c,i}}\right)\left[1+(0.37464+1.54226\omega_i-0.26992\omega_i^2)\left(1-\left(\frac{T}{T_{c,i}}\right)^{1/2}\right)\right]^2 \\ b_i&=0.07780\frac{RT_{c,i}}{P_{c,i}} \end{aligned} \tag{12.181}$$

where $T_{c,i}$ is the critical temperature, $p_{c,i}$ is the critical pressure and ω_i is the acentric factor of the component i.

The fugacities of the components depend on the compressibility of the liquid and vapor phase:

$$ln\varphi_i=\left(\frac{\partial\left(A^\gamma/RT\right)}{\partial N_i}\right)_{T,V,N_{j\neq i}}-lnZ \tag{12.182}$$

where A^γ is the residual Helmholtz energy, which is a function of the compressibility:

$$\frac{A^\gamma}{RT}=\int_V^\infty\left[1-Z\right]\frac{dV}{V}+lnZ \tag{12.183}$$

In summary, the vapor mole fraction x_i^V, the pressure p, and the compressibilities of the vapor (Z^V) and the liquid (Z^L) phase at the surface of the particle are determined from the liquid particle mole fraction of the components x_i^L and the particle temperature T_p. The surface vapor concentrations are calculated using the following equation:

$$C_{i,S}=x_i^V\frac{p}{Z^VRT} \tag{12.184}$$

12.6. Physical Property Averaging

The droplet evaporation and heating rates are very sensitive to the physical property values used in the rate equations, so it is important to use appropriate temperature and composition assumptions for property evaluation. The most common method for evaluating properties is to compute averaged

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

properties $\overline{\varphi}$ based on reference values for the temperature $\overline{T}$ and vapor mass fraction $\overline{Y}$ as follows ([439] (p. 1078), [7] (p. 1053), and [550] (p. 1085)):

$$\overline{T}=T_s+\alpha_T\left(T_g-T_s\right) \tag{12.185}$$

$$\overline{Y}=Y_s+\alpha_Y\left(Y_g-Y_s\right) \tag{12.186}$$

where,

$\overline{T}$ = reference temperature

$\overline{Y}$ = reference vapor mass fraction

T_g and T_s = temperatures at far field conditions and at the droplet surface, respectively

Y_g and Y_s = evaporating species mass fraction in the far field gas mixture and at the droplet surface, respectively

α_T and α_Y = averaging coefficient for temperature and mass fraction, respectively

This averaging method is known as the 1/3 rule because of the most commonly used value for the parameters α = 1/3.

Assuming that the species mass fractions in the reference mixture are proportional to the mass fractions in the far field gas mixture, except for the evaporating species, the mass averaged property $\overline{\varphi}$ can be obtained from:

$$\overline{\varphi}=\overline{Y}\overline{\varphi_v}+\frac{\left(1-\overline{Y}\right)}{\left(1-Y_g\right)}\left(\overline{\varphi_g}-Y_g\overline{\varphi_v}\right) \tag{12.187}$$

where $\overline{\varphi_g}$ and $\overline{\varphi_v}$ are the vapor and far field gas mixture properties at the reference temperature $\overline{T}$, that is:

$$\overline{\varphi_v}=\varphi_v\left(\overline{T}\right) \tag{12.188}$$

$$\overline{\varphi_g}=\varphi_g\left(\overline{T}\right) \tag{12.189}$$

If the evaporating species mass fraction in the bulk Y_g is 0, Equation 12.187 (p. 491) reduces to the simple mass averaging formula:

$$\overline{\varphi}=\overline{Y}\overline{\varphi_v}+\left(1-\overline{Y}\right)\overline{\varphi_g} \tag{12.190}$$

The averaging coefficients are assigned values between 0 and 1. As follows from equations Equation 12.185 (p. 491) and Equation 12.186 (p. 491), α= 1 corresponds to free stream conditions, while for α = 0, the temperature and composition are those of the particle surface.

For multicomponent evaporation, the averaging formula takes the form:

$$\overline{\varphi}=\sum_{all\ v_i}\overline{Y_{v_i}}\overline{\varphi_{v_i}}+\frac{\left(1-\sum_{all\ v_i}\overline{Y_{v_i}}\right)}{\left(1-\sum_{all\ v_i}Y_{g_i}\right)}\left[\overline{\varphi_g}-\sum_{all\ v_i}\left(Y_{g_i}\overline{\varphi_{v_i}}\right)\right] \tag{12.191}$$

where the subscript v_i refers to the vapor of the i^{th} evaporating species, and the subscript g_i refers to the i^{th} species in the far field gas mixture.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

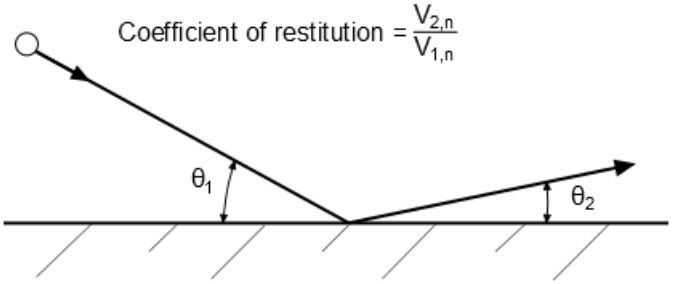
The property averaging equations Equation 12.187 (p. 491) and Equation 12.191 (p. 491) are applied for all physical properties in the droplet and multicomponent rate equations, respectively, except averaged density $\overline{\rho}$, which is computed for temperature $\overline{T}$ and mass fractions $\overline{Y}$ directly from the density property model equation defined for the gas phase mixture material:

$$\overline{\rho}=\rho_g\left(\overline{Y},\overline{T}\right) \tag{12.192}$$

12.7. Wall-Particle Reflection Model Theory

For non-rotating particles, Ansys Fluent provides a model that accounts for the energy loss due to inelastic collision with the wall. The particle rebounds off the boundary in question with a change in its momentum as defined by the coefficient of restitution. (See Figure 12.2: Particle Reflection at Wall (p. 492).)

Figure 12.2: Particle Reflection at Wall



The normal coefficient of restitution defines the amount of momentum in the direction normal to the wall that is retained by the particle after the collision with the boundary [639] (p. 1090):

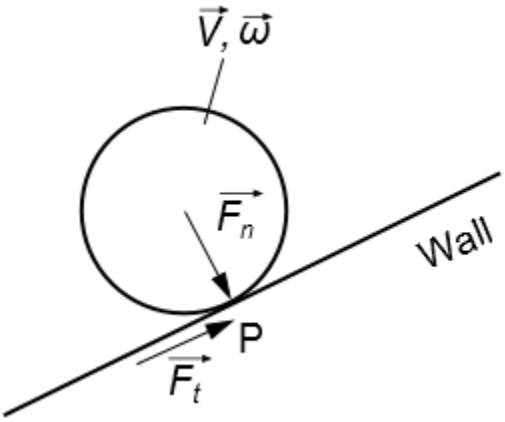
$$e_n=\frac{V_{2,n}}{V_{1,n}} \tag{12.193}$$

where V_n is the particle velocity normal to the wall and the subscripts 1 and 2 refer to before and after collision, respectively. Similarly, the tangential coefficient of restitution, e_t, defines the amount of momentum in the direction tangential to the wall that is retained by the particle.

For rotating particles, Ansys Fluent provides a model for estimating a particle velocity after rebounding from a wall, as proposed by Tsuji et al. [657] (p. 1091). The model is based on classical mechanics (see Figure 12.3: Particle-Wall Collision Forces (p. 493)) and accounts for energy loss due to inelastic collision and friction.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Figure 12.3: Particle-Wall Collision Forces



In this model, the time during which the particle and the wall are in contact is divided into two periods; namely, compression and recovery. The system of equations is defined according to whether or not the particle stops sliding within the compression period.

Adhesive Impact

If the particle stops sliding within the compression period, the following condition holds:

$$F_t \leq \mu F_n \tag{12.194}$$

The limiting case of equality in Equation 12.194 (p. 493) corresponds to the Coulomb friction law (Equation 12.505 (p. 580)), where the tangential force F_t is directly proportional to the normal force F_n, and the proportionality factor is the friction coefficient μ.

The momentum equations for the particle motion before and after the collision with the wall are as follows:

$$\begin{aligned} m_p\left(V_n - V_n^0\right) &= -J_n \\ m_p\left(V_t - V_t^0\right) &= -J_t \\ I_p\left(\omega_n - \omega_n^0\right) &= M_n \\ I_p\left(\omega_t - \omega_t^0\right) &= M_t = -r \times J_t \end{aligned} \tag{12.195}$$

where,

m_p and I_p are the particle mass and moment of inertia, respectively (the latter assumes spherical particles)

M is the torque with respect to the center of mass

V and ω are the linear and angular particle velocities, respectively

J is the momentum

r is the particle radius

the superscript 0 refers to particle velocities before the collision with the wall

the subscripts n and t denote the normal and tangential components of the variables, respectively

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

The coefficient of restitution e_n is defined as the ratio of the momenta in the normal direction before and after the collision. If the particle mass remains constant during rebounding off the wall, the coefficient of restitution can be simply defined as the ratio of the particle normal velocities before and after the collision:

$$e_n = \frac{J_n}{J_n^0} = -\frac{V_n}{V_n^0} \tag{12.196}$$

The angular momentum in the wall normal direction is not affected by the collision, that is $M_n = 0$. This results in static (unchanging) particle angular velocity:

$$\omega_n = \omega_n^0 \tag{12.197}$$

For adhesive impact, the following boundary condition for the particle velocity at the point of contact with the wall can be defined:

$$V_t^p = V_t + r \times \omega_t = 0 \tag{12.198}$$

where the superscript p refers to the particle velocity at the contact point during collision (see Figure 12.3: Particle-Wall Collision Forces (p. 493)).

For a given boundary condition, the post-collision velocities, V and ω, can be computed by solving Equation 12.195 (p. 493).

Sliding Impact

If the tangential force exceeds the frictional force (that is, the inequality in Equation 12.194 (p. 493) is not fulfilled), the particle does not stop sliding within the compression period. The particle linear and angular velocities are not correlated kinematically, and post-collision velocities can be estimated directly from the respective momentum equations. Equation 12.196 (p. 494) and Equation 12.197 (p. 494) are still valid. The change in the tangential momentum can be defined using Equation 12.194 (p. 493) as follows:

$$m_p\left(V_t - V_t^0\right) = \int_{t1}^{t0} F_t dt = \int_{t1}^{t0} -\mu \ F_n \varepsilon_t dt \tag{12.199}$$

where,

$$\varepsilon_t = sign\left(V_t\right) \tag{12.200}$$

where $sign$ is the sign function that returns +1 for positive values and –1 for negative values.

Note:

When particle rotation is enabled, the tangential coefficient of restitution (the **DPM** tab of the **Wall** boundary condition dialog box) is not taken into account. The tangential linear velocity of a particle after rebounding from the wall is a function of both the particle's linear velocity V and its angular velocity ω before the collision.

See Particle Rotation in the *Fluent User's Guide* for information about how to use the particle rotation model.

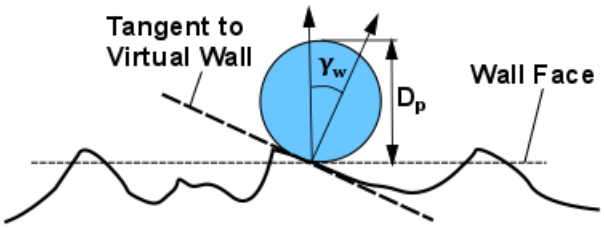
Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

12.7.1. Rough Wall Model

The rough wall model can capture more realistic behavior of dispersed flows in confined geometries, such as cyclones and pipe systems. Studies on particle-wall interaction have shown that the motion of large or heavy particles is strongly influenced by the surface roughness.

The rough wall model used in Ansys Fluent is based on the theoretical findings in [614] (p. 1088) and [189] (p. 1063). At the moment of particle-wall collision, the wall face is replaced by a face that is tangential to the virtual rough wall at the point of particle-wall contact.

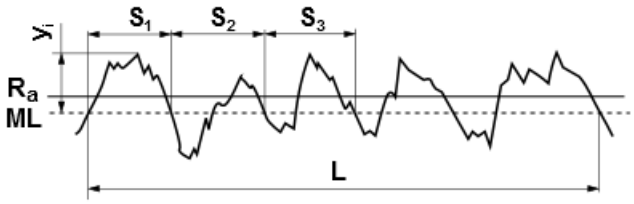
Figure 12.4: Particle-Wall Interaction at a Rough Wall



The inclination angle γ_w (that is, the angle between the normal vectors of the wall face and the face that is tangential to the virtual wall), is computed as a function of the statistical surface roughness parameters and the current particle diameter.

Figure 12.5: Wall Roughness Parameters (p. 495) illustrates how the surface roughness parameters are estimated. For detailed information on statistical surface roughness parameters used in industry, see ISO 4287.

Figure 12.5: Wall Roughness Parameters



ML = mean line
$\mathbf{y_i}$ = surface height (with respect to ML at i^{th} sampling position)
L = sampling length
$\mathbf{S_i}$ = profile element length

The wall roughness parameters are:

- Wall mean roughness R_a:

$$R_a = \frac{1}{L}\int_0^L |y(x)|dx$$

- Standard deviation of the roughness structure R_q:

$$R_q=\sqrt{\frac{1}{L}\int_0^L y^2(x)dx}$$

- Mean profile element width/spacing within a sampling length RS_m:

$$RS_m=\frac{1}{p}\sum_{i=1}^{p}S_i$$

  where p is the number of profile elements in sampling length L.

The inclination angle γ_w is sampled from a Gaussian distribution with mean value of 0° and a standard deviation of $\Delta\gamma$.

The standard deviation of the Gaussian distributed virtual wall angle is estimated by:

$$\Delta\gamma=\begin{cases}\arctan\left(2R_q/RS_m\right) & if\ \ D_p<\frac{RS_m}{\sin\left(\arctan\left(2R_a/RS_m\right)\right)}\\ \arctan\left(2R_a/RS_m\right) & otherwise\end{cases} \tag{12.201}$$

where,

$\Delta\gamma$ = standard deviation of the Gaussian distributed virtual wall angle

D_p = particle diameter

The model accounts for the "shadow effect" [614] (p. 1088) in which particles cannot physically collide with a virtual wall if the angle between the particle velocity vector (prior to the collision) and the normal vector of the virtual wall is greater than $\pi/2$ rad.

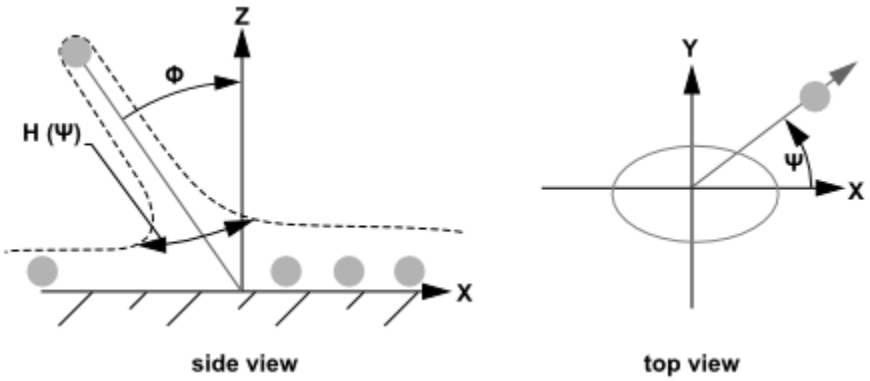
Once the new virtual wall is defined by its normal vector, Equation 12.193 (p. 492) to Equation 12.200 (p. 494) are used to compute the rebound velocity of the reflected particle with respect to the virtual wall.

12.8. Wall-Jet Model Theory

The **wall-jet** boundary condition provides a range of rebound directions and velocities when a liquid droplet collides with a wall. It is suitable for situations where droplets impact a hot wall, where no liquid film is formed, but the droplets reflect or stick on the wall depending on their impact properties. The direction and velocity of the droplet particles are given by the resulting momentum flux, which is a function of the impingement angle, φ, and Weber number. See Figure 12.6: "Wall Jet" Boundary Condition for the Discrete Phase (p. 497).

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Figure 12.6: "Wall Jet" Boundary Condition for the Discrete Phase



The **wall-jet** type boundary condition assumes an analogy with an inviscid jet impacting a solid wall. Equation 12.202 (p. 497) shows the analytical solution for an axisymmetric impingement assuming an empirical function for the sheet height (H) as a function of the angle that the drop leaves the impingement (Ψ).

$$H(\Psi)=H_{\pi}e^{\beta\left(1-\frac{\Psi}{\pi}\right)} \tag{12.202}$$

where H_{π} is the sheet height at $\Psi=\pi$ and β is a constant determined from conservation of mass and momentum. The probability that a drop leaves the impingement point at an angle between Ψ and $\Psi+\delta\Psi$ is given by integrating the expression for $H(\Psi)$

$$\Psi=-\frac{\pi}{\beta}\ln\left[1-P\left(1-e^{-\beta}\right)\right] \tag{12.203}$$

where P is a random number between 0 and 1. The expression for β is given in Naber and Reitz [465] (p. 1080) as

$$\sin(\varphi)=\frac{e^{\beta}+1}{\left(e^{\beta}-1\right)\left(1+\left(\frac{\pi}{\beta}\right)^{2}\right)} \tag{12.204}$$

Fluent uses a simplified expression for computing β from Equation 12.204 (p. 497).

12.9. Lagrangian Wall-Film Model Theory

This section is composed of the following:

12.9.1. Introduction
12.9.2. Leidenfrost Temperature Considerations
12.9.3. Interaction During Impact with a Boundary
12.9.4. Separation and Stripping Submodels
12.9.5. Conservation Equations for Wall-Film Particles

12.9.1. Introduction

Ansys Fluent has a specific boundary condition for simulation of liquid droplets colliding with walls and forming thin films. This model is particularly aimed at simulation of internal combustion engines. The Lagrangian wall film model uses transient Lagrangian particles to discretize physical effects and

is therefore well suited for fast transient effects. Besides this, an Eulerian wall film model exists that is able to deal with steady wall films (Eulerian Wall Films (p. 839)).

Spray-wall interaction is an important part of the mixture formation process in port fuel injected (PFI) engines. A fuel spray impinges on a surface, usually at the intake port near the intake valve, as well as at the intake valve itself, where it splashes and subsequently evaporates. The evaporated mixture is entrained into the cylinder of the engine, where it is mixed with the fresh charge and any residual gas in the cylinder. The mixture that is compressed and burned, finally exits through the exhaust port. The process repeats itself between 200 and 8000 times per minute, depending on the engine.

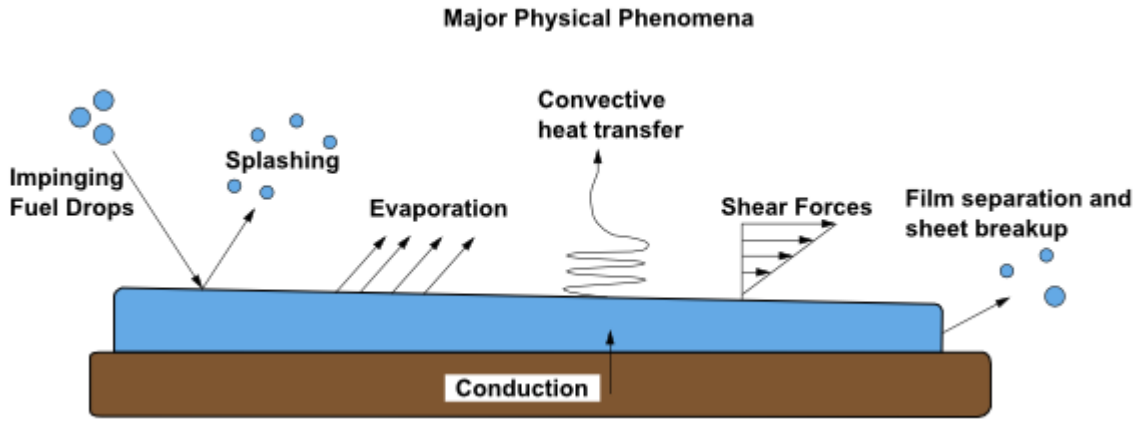
Several cycles worth of fuel remain in the intake tract due to film formation on the walls. This in turn makes the film important in hydrocarbon emissions for PFI engines. Additionally, film can form inside combustion chambers of direct injection (DI) types of engines. In a direct injection engine, fuel is injected directly into the combustion chamber, where the spray can impinge upon the piston if the injection event is early or late in the cycle. The modeling of the wall-film inside a DI engine, especially in diesel engines, is compounded by the presence of carbon deposits on the surfaces of the engine. This carbon deposit absorbs the liquid film as it impinges upon it. It is believed that the carbon deposits adsorb the fuel later in the cycle, however this phenomenon is very complex and is not well understood.

In addition, wall film formation is an important factor for the Selective Catalytic Reduction (SCR) aftertreatment process. The SCR of oxides of nitrogen with ammonia is a well-established emission control technology for meeting stringent NO_x emission standards across the world. SCR system designers face several technical challenges, such as ammonia slip, urea crystallization, low temperature deposits, and other potential pitfalls. By not including the physics of wall film formation in SCR predictive models, large errors can be introduced in the NO_x conversion predictions.

Finally, wall film modeling plays an important role in gas turbine operation and cooling of electronics.

DPM particles are used to model the wall-film. The wall-film model in Ansys Fluent allows a liquid drop to impinge upon a boundary surface and form a thin film. The model can be broken down into four major subtopics: interaction during the initial impact with a wall boundary, subsequent tracking on surfaces, calculation of film variables, and coupling to the gas phase. Figure 12.7: Mechanisms of Splashing, Momentum, Heat and Mass Transfer for the Lagrangian Wall-Film (p. 498) schematically shows the basic mechanisms considered for the Lagrangian wall-film model.

Figure 12.7: Mechanisms of Splashing, Momentum, Heat and Mass Transfer for the Lagrangian Wall-Film



Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

The main assumptions and restrictions for the Lagrangian wall-film model are as follows:

- The layer is thin, less than 500 microns in thickness. This limitation is due to the assumption of a linear velocity profile in the film.
- The temperature in the film particles change relatively slowly so that an analytical integration scheme can be utilized.
- Film particles are assumed to be in direct contact with the wall surface and the heat transfer from the wall to the film takes place by conduction.
- The film temperature never exceeds the boiling temperature for the liquid.

If you want to model a spray impacting a very hot wall, the wall-jet model may be more appropriate as the assumption in the wall-jet impingement model is that there is a vapor layer underneath the drops that keeps them from making direct contact with the boundary surface. This may be a more accurate assumption for in-cylinder diesel computations at typical operating conditions.

For more information about using the Lagrangian wall-film model in Ansys Fluent, see Modeling Lagrangian Wall Films in the *Fluent User's Guide*.

12.9.2. Leidenfrost Temperature Considerations

In the Lagrangian Wall Film (LWF) model, the wall film vaporization rate equation (Equation 12.275 (p. 519) for the diffusion controlled model, Equation 12.282 (p. 520) for the convection-diffusion controlled model, and Equation 12.291 (p. 523) and Equation 12.292 (p. 523) for the film boundary layer model) assumes that the liquid film is in full contact with the wall. When the wall film temperature reaches the boiling point, a boiling rate is computed; however, the assumption still is that the liquid film fully covers the wall surface.

The LWF boiling rate equation (Equation 12.283 (p. 520) and Equation 12.284 (p. 521) for constant-temperature walls and heat-flux walls, respectively) gives a linear dependence of the boiling rate on the wall temperature. However, liquid film boiling phenomena are much more complex, and the rate of heat transfer from the hot wall to the film deviates from the linear dependence as the temperature difference between the wall and the liquid film increases. This deviation is caused by vapor pockets that develop between the hot wall surface and the boiling film, which significantly affect both the heat transfer and the hydrodynamic characteristics of the film boiling process. The limiting condition in which the insulating vapor layer fully covers the hot wall is known as the Leidenfrost point.

Fluent deals with wall surface boiling phenomena in the context of the multiphase models, in particular in Wall Boiling Models (p. 733). However, in the context of the LWF model, the prediction of the Leidenfrost point is important for assessing the adequacy of the LWF boiling rate and avoiding unphysical results.

12.9.2.1. Default Wall Temperature Limiter

In Ansys Fluent, a default limit for the wall temperature T_{w_lim} is a sum of the liquid film boiling point T_{boil} and the temperature difference between the wall and the liquid film ΔT:

$$T_{w_lim} = \left(T_{boil} + \Delta T \right) \tag{12.205}$$

where the default ΔT is 50 K.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

12.9.2.2. Leidenfrost Temperature Reporting

During the LWF computation, Ansys Fluent calculates the Leidenfrost temperature and compares it to the wall temperature. If the wall temperature is above the Leidenfrost temperature, Fluent issues a warning. Note that the calculation of the Leidenfrost point is for information only and is not used to limit the wall temperature in the calculation.

For an isothermal surface, the Leidenfrost temperature $T_{Leid,\,iso}$ can be approximated as a function of the liquid critical temperature T_{crit} and the liquid surface tension σ [51] (p. 1055):

$$T_{Leid,\,iso} = \frac{27}{32} T_{crit} \left(1 - \exp\left(-0.52 \left[\frac{10^4 \left(\rho / A\right)^{4/3}}{\sigma}\right]^{1/3}\right)\right) \tag{12.206}$$

where ρ is the solid density, and A is the solid atomic number. The last term in Equation 12.206 (p. 500) introduces a correction for the solid-liquid surface tension effects.

Knowing $T_{Leid,\,iso}$ and the heat transfer properties of the solid surface, the Leidenfrost temperature T_{Leid} can be calculated as:

$$\frac{T_{Leid,\,iso} - T_f}{T_{Leid} - T_f} = \exp\left(0.00175\,\beta\right) \operatorname{erfc}\left(0.042\sqrt{\beta}\right) \tag{12.207}$$

where T_f is liquid film temperature (K), erfc is the complementary error function, and

$$\beta = \frac{1}{k \rho c_p}$$

where

k = solid thermal conductivity (W/m-K)

ρ = solid density (kg/m^3)

c_p = solid heat capacity (J/kg-K)

12.9.3. Interaction During Impact with a Boundary

Predicting drop/wall impingement is not straightforward because of the large number of parameters involved. The outcome of any impingement is governed by the properties of the droplet liquid (viscosity, surface tension, and density) and by the impingement conditions (droplet diameter and droplet velocity). The two other important factors that contribute to the impingement outcome are the surface roughness and the thickness of the liquid film on the surface. In addition, wall temperature may also have an effect on impinging dynamics.

There are several models that represent impingement physics and predict the impingement outcome. Ansys Fluent provides the following models:

- The Stanton and Rutland model [626] (p. 1089), [628] (p. 1089) uses the wall temperature and a Weber number-based criterion to define fixed value thresholds between the impingement regimes (stick, rebound, spread, and splash). In Fluent implementation, the original Stanton-Rutland model has been extended using the concepts from O'Rourke and Amsden [492] (p. 1081). See The Stanton-Rutland Model (p. 501) for more details.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

- The Kuhnke model [322] (p. 1071) considers both dry and wet walls and takes into account, in addition to the drop properties and impingement conditions, the influence of surface roughness and film thickness. The Kuhnke model distinguishes between the following regimes: rebound, spread, splash, and dry splash (thermal breakup). For more information, see The Kuhnke Model (p. 508).
- The Stochastic Kuhnke model [603] (p. 1088), [37] (p. 1055) is derived from the Kuhnke model and has the following features:
  - Uses different regime transition criteria that have been developed and tuned for Selective Catalytic Reduction (SCR) modeling applications.
  - Includes stochastic effects into the critical temperature transition process.
  - Introduces the evaporative splash regime with the "partial evaporation" concept, where a fraction of the impinging droplets is assumed to completely vaporize under certain conditions.

    See The Stochastic Kuhnke Model (p. 514) for details.
- The user-defined Impingement Model can be employed through the use of `DEFINE_IMPINGEMENT`, `DEFINE_FILM_REGIME` and `DEFINE_SPLASHING_DISTRIBUTION` user-defined functions as described in User-Defined Functions in the *Fluent User's Guide* and in the Fluent Customization Manual.

Note:

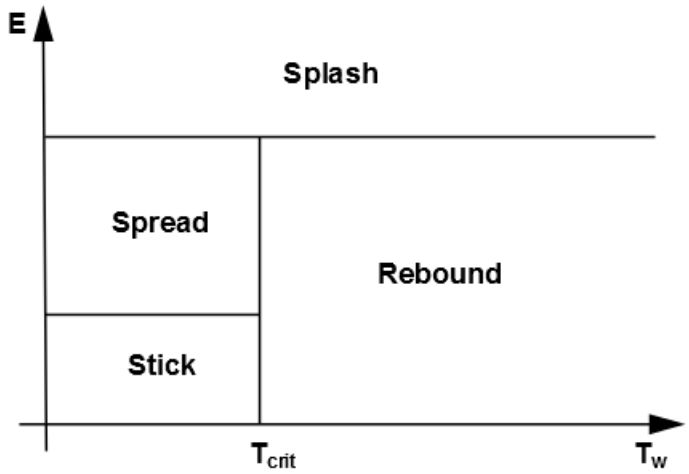
- The Lagrangian wall film model is available for the cases where energy equation is not solved with the below assumptions for the impingement/splashing models:
  - The wall temperature is 300 K
  - The saturation temperature of the liquid film is equal to 5000 K
- The particle velocities referred in the sections that follow are relative to the frame of the wall. Dimensionless parameters Re and We are based on particle velocities normal to the wall unless specified.

12.9.3.1. The Stanton-Rutland Model

12.9.3.1.1. Regime Definition

The wall interaction is based on the work of Stanton [626] (p. 1089) and O'Rourke [493] (p. 1081), where the regimes are calculated for a drop-wall interaction based on local information. The four regimes (stick, rebound, spread, and splash) are based on the impact energy and wall temperature. The following chart is helpful in showing the cutoffs.

Figure 12.8: Simplified Decision Chart for Wall Interaction Criterion



Below the critical transition temperature of the liquid, the impinging droplet can either stick, spread or splash, while above the critical transition temperature, the particle can either rebound or splash.

The criteria by which the regimes are partitioned are based on the impact energy and the critical transition temperature of the liquid. The impact energy is defined by

$$E^2=\frac{\rho V_{pn}^2 d_p}{\sigma}\left(\frac{1}{\min\left(h_0/\ d_p,1\right)+\delta_{bl}/\ d_p}\right) \tag{12.208}$$

where V_{pn} is particle velocity normal to the wall, ρ is the liquid density, d_p is the diameter of the droplet, σ is the surface tension of the liquid, and h_0 is the film height. Here, δ_{bl} is a boundary layer thickness, defined by

$$\delta_{bl}=\frac{d_p}{\sqrt{Re}} \tag{12.209}$$

where the Reynolds number is defined as $Re=\rho V_{\mathrm{pn}} d_p/\ \mu$. By defining the energy as in Equation 12.208 (p. 502), the presence of the film on the wall suppresses the splash, but does not give unphysical results when the film height approaches zero.

The critical transition temperature is defined as:

$$\mathrm{T_{crit}} = \mathrm{T^*_{crit}}\mathrm{T_{sat}} \tag{12.210}$$

where $\mathrm{T_{sat}}$ is the saturation temperature, and $\mathrm{T^*_{crit}}$ is the critical temperature factor with values between 1 and 1.5 depending on the application. The default value for $\mathrm{T^*_{crit}}$ is 1; as a result, the default critical transition temperature $\mathrm{T_{crit}}$ will be equal to the saturation temperature of the droplet liquid $\mathrm{T_{sat}}$.

As an alternative for the Lagrangian wall film model, the critical transition temperature T_{crit} can be calculated from the saturation temperature of the liquid T_{sat} by adding a constant offset T_{cal}:

$$T_{crit}=T_{sat}+T_{cal} \tag{12.211}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

The sticking regime is applied when the dimensionless energy E is less than 16, and the particle velocity is set equal to the wall velocity. In the spreading regime, the initial direction and velocity of the particle are set using the wall-jet model, where the probability of the drop having a particular direction along the surface is given by an analogy of an inviscid liquid jet with an empirically defined radial dependence for the momentum flux.

If the wall temperature is above the critical transition temperature of the liquid, impingement events below a critical impact energy (E_{cr}) results in the particles rebounding from the wall.

Splashing occurs when the impingement energy is above a critical energy threshold, defined as $E_{cr}=57.7$. The number of splashed droplet parcels is set in the **Wall** boundary condition dialog box with a default number of 4, however, you can select numbers between zero and ten. The splashing algorithm follows that described by Stanton [626] (p. 1089) and is detailed in Splashing (p. 503).

12.9.3.1.2. Rebound

In the rebound regime, the velocity of the rebounding particle is given by:

$$V_{rt}=\frac{5}{7}V_{pt} \tag{12.212}$$

$$V_{rn}=e_nV_{pn} \tag{12.213}$$

where V_{rt} and V_{rn} are the tangential and normal components of the rebounding velocity, V_{pt} is the tangential component of the impinging velocity, and e_n is the normal restitution coefficient given by:

$$e_n=0.993-1.76\Theta_I+1.56\Theta_I^2-0.49\Theta_I^3 \tag{12.214}$$

where Θ_I is the impingement angle in radians as measured from the wall surface.

The deviation angle of the rebounding particle, ψ, takes a random value between -90° and 90° from the incident direction.

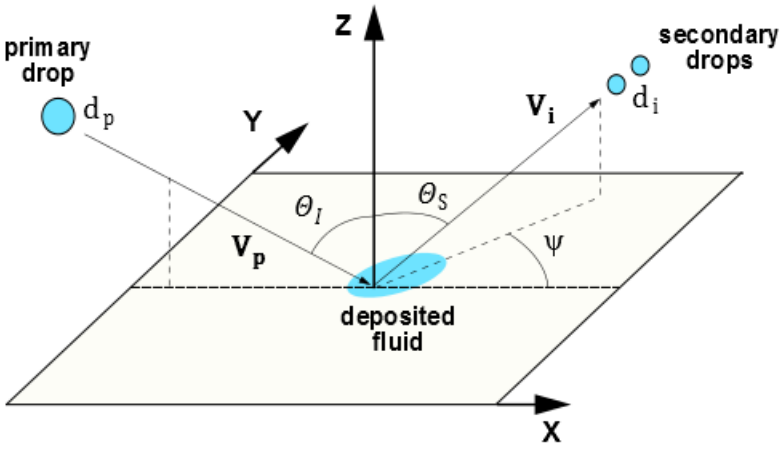
12.9.3.1.3. Splashing

Figure 12.9: The Stanton-Rutland Model: Impinging and Splashing (p. 504) graphically depicts the following splashed droplet variables for the Stanton-Rutland splashing model.

- Diameter of the impinging particle, d_p
- Impinging particle velocity, V_p plus variability
- Diameters of the splashed secondary particles, d_i
- Velocities of the splashed secondary particle V_i
- Impingement angle Θ_I measured from the wall surface normal
- Reflection angle of the splashed particle Θ_s measured from the wall surface normal
- Deviation angle ψ from the incident direction

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Figure 12.9: The Stanton-Rutland Model: Impinging and Splashing



If the particle impinging on the surface has a sufficiently high energy, the particle splashes and several new particles are created. You can explicitly set the number of particles created by each impact in the **DPM** tab or the **Wall Film** tab of the **Wall** boundary condition dialog box. The minimum number of splashed parcels is three. The properties (diameter, magnitude, and direction) of the splashed parcels are randomly sampled from the experimentally obtained distribution functions described in the following sections. Setting the number of splashed parcels to zero in the Lagrangian wall film model turns off the splashing calculation for the Lagrangian model. Bear in mind that each splashed parcel can be considered a discrete sample of the distribution curves and that selecting the number of splashed drops in the **Wall** boundary condition dialog box does not limit the number of splashed drops, only the number of *parcels* representing those drops.

Therefore, for each splashed parcel, a different diameter is obtained by sampling a cumulative probability distribution function (CPDF), also referred to as F, which is obtained from a Weibull distribution function and fitted to the data from Mundo, et al. [462] (p. 1080). The equation is

$$f(d) = 2\frac{d}{D^2}\exp\left[-\left(\frac{d}{D}\right)^2\right] \tag{12.215}$$

and it represents the probability of finding drops of diameter d_i in a sample of splashed drops. This distribution is similar to the Nakamura-Tanasawa distribution function used by O'Rourke [493] (p. 1081) with distribution parameter $D=\sqrt{2}\,d_{max}$. To ensure that the distribution functions produce physical results with an increasing Weber number, the following expression for d_{max} from O'Rourke [493] (p. 1081) is used. The ratio of the particle diameter at the peak of the splashed size distribution to the impinging particle diameter is given by

$$d_{max}/\,d_p = \max\left(\frac{E_{crit}^2}{E^2}, \frac{6.4}{We}, 0.06\right) \tag{12.216}$$

where the expression for energy is given by Equation 12.208 (p. 502). Low Weber number impacts are described by the second term in Equation 12.216 (p. 504) and the peak splashed diameter ratio is never less than 0.06 for very high energy impacts in any of the experiments analyzed by O'Rourke [493] (p. 1081). The Weber number in Equation 12.216 (p. 504) is defined using normal particle velocity and diameter:

$$We = \frac{\rho V_{pn}^2 d_p}{\sigma} \tag{12.217}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

The cumulative probability distribution function (CPDF) is needed so that a diameter can be sampled from the experimental data. The CPDF is obtained from integrating Equation 12.215 (p. 504) to obtain

$$F(d) = 1 - \exp\left[-\left(\frac{d}{D}\right)^2\right] \tag{12.218}$$

which is bounded by zero and one. An expression for the diameter (which is a function of D, the impingement Weber number We, and the impingement energy) is obtained by inverting Equation 12.218 (p. 505) and sampling the CPDF between zero and one. The expression for the diameter of the i^{th} splashed parcel is therefore given by $d_i = D\sqrt{-\ln(1-c_i)}$ where c_i is the i^{th} random sample. Once the diameter of the splashed drop has been determined, the probability of finding that drop in a given sample is determined by evaluating Equation 12.215 (p. 504) at the given diameter.

In order to prevent the splashed particle diameter from becoming too large, Ansys Fluent uses distribution-limiting methods:

- The tail limiting method (default for the Lagrangian Wall Film model) restricts the ratio

  $$d_{99.9} / d_p < 1.0$$

  where $d_{99.9}$ is the particle diameter at 99.9% of CPDF.

- The peak limiting method (default for the Eulerian Wall Film model) restricts the ratio

  $$d_{max} / d_p < P_{max}$$

  where P_{max} is a limiting value ranging from 0.2 to 0.3 (default 0.22) as determined from the experimental data by Mundo [462] (p. 1080).

The limiting method and parameter P_{max} can be selected through the text user interface (TUI) commands.

The number of droplets in each parcel is proportional to the value of the PDF for that droplet size. The total number of particles is then determined by the total mass that is splashed.

The mass fraction of the splashed particle y_s is obtained from the experimental data by Yarin and Weiss [721] (p. 1095). The splashed mass fraction expression by O'Rourke et al. [493] (p. 1081) is given by:

$$y_s = \begin{cases} 1.8x10^{-4}\left(E^2 - E_{crit}^2\right), & E_{crit}^2 < E^2 < 7500 \\ 0.75, & E^2 > 7500 \end{cases} \tag{12.219}$$

The splash mass fraction expression from Stanton et al. [626] (p. 1089) is given by:

$$y_s = \begin{cases} 0.0 & K_u \le 18.0 \\ -27.2 + 3.15K_u - 0.116K_u^2 + 1.40\times10^{-3}K_u^3 & 18.0 < K_u < 33.0 \\ 0.75 & K_u \ge 33.0 \end{cases} \tag{12.220}$$

where K_u is the non-dimensional impact velocity defined as:

$$K_u = G\sqrt{K} \tag{12.221}$$

with K defined by Equation 12.236 (p. 508) and the constant G taking the value of 2.37.

The splash mass fraction method can be selected through the TUI command, with the O'Rourke expression as default for Lagrangian Wall Film (LWF) and the Stanton expression as default for the Eulerian Wall Film (EWF).

The authors (O'Rourke et al. [493] (p. 1081)) noted that nearly all of the impacts for typical diesel sprays are well above the upper bound and so the splashing event nearly always ejects 75% of the mass of the impinging drop. To obtain an expression for the total number of drops, we note that overall conservation of mass requires that the sum of the total mass of the splashed parcel(s) must equal the splashed mass fraction, or

$$\frac{\rho\pi}{6} N_{tot} \sum_{i=1}^{N_{parcels}} \left(f_i d_i^3 \right) = y_s m_p \tag{12.222}$$

where m_p is the total mass of the impinging parcel. The expression for the total number of splashed drops is

$$N_{tot} = y_s m_p \Big/ \left(\frac{\rho\pi}{6} \sum_{i=1}^{N_{parcels}} \left(f_i d_i^3 \right) \right) \tag{12.223}$$

The number of splashed drops per parcel is then determined by N_{tot} multiplied by the value of the PDF for that droplet size.

To calculate the velocity with which the splashed drops leave the surface, additional correlations are sampled for the normal component of the velocity. A Weibull function, fit to the data from Mundo [462] (p. 1080), is used as the PDF for the normal component. The probability density is given by

$$f\left(\frac{V_{in}}{V_{pn}} \right) = \left[\frac{b_v}{\Theta_v} \left(\frac{V_{in} / V_{pn}}{\Theta_v} \right)^{b_v - 1} \right] \exp\left[- \left(\frac{V_{in} / V_{pn}}{\Theta_v} \right)^{b_v} \right] \tag{12.224}$$

where the index i refers to the i^{th} splashed droplet, and the subscript p refers to the impinging droplet. The terms b_v and Θ_v are functions of Θ_I, the impingement angle measured from the surface normal:

$$b_v = \begin{cases} 2.1, & \Theta_I \le 50^\circ \\ 1.10 + 0.02\Theta_I, & \Theta_I > 50^\circ \end{cases}$$

$$\Theta_v = 0.158 e^{0.017\Theta_I}$$

The tangential component of the velocity is obtained from the expression for the reflection angle Θ_s:

$$\Theta_s = 65.4 + 0.226\Theta_I \tag{12.225}$$

combined with

$$V_{it} = V_{in} \tan\left(\Theta_s \right) \tag{12.226}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Note that the reflection angle Θ_s is measured from the surface normal, and both Θ_s and Θ_I are in degrees.

The azimuthal angle ψ for the splashed droplets is computed according to [627] (p. 1089) assuming that it follows the distribution $H(\psi)$ (Equation 12.227 (p. 507)). The treatment is based on the approach developed by Naber and Reitz [465] (p. 1080).

$$H(\psi)=\begin{cases}1/2\pi & \beta=0\\ \frac{\beta/\pi}{2(e^{\beta}-1)}\exp[\beta(1-(|\psi|/\pi))] & \beta>0\end{cases} \tag{12.227}$$

where β is a parameter related to the impingement angle Θ_I by:

$$\sin\Theta_I=\frac{e^{\beta}+1}{e^{\beta}-1}\frac{\beta^2}{\beta^2+\pi^2} \tag{12.228}$$

ψ is determined by integrating Equation 12.227 (p. 507) and Equation 12.228 (p. 507) as follows:

- For β=0:

$$\psi=2\pi p \tag{12.229}$$

- For β>0:

$$\psi=\begin{cases}(\pi/\beta)\log[e^{-\beta}+2p(1-e^{-\beta})] & for\ p<0.5\\ (\pi/\beta)\log[2-e^{-\beta}-2p(1-e^{-\beta})] & for\ p>0.5\end{cases} \tag{12.230}$$

In the above equations, p is a uniformly distributed random number between 0 and 1:

$$p=rnd(0,1)$$

Finally, an energy balance is performed for the new parcels so that the sum of the kinetic and surface energies of the new drops does not exceed that of the old drops. The energy balance is given by

$$\frac{1}{2}\sum_{i=1}^{N_{parcel}}(m_iV_i^2)+\pi\sigma\sum_{i=1}^{N_{parcel}}(N_id_i^2)=\frac{1}{2}m_pV_{pn}^2+\pi\sigma(N_pd_p^2)-E_s \tag{12.231}$$

where E_s is the threshold energy for splashing to occur. To ensure conservation of energy, the following correction factor is computed:

$$E_k=\frac{\frac{1}{2}m_pV_{\mathrm{pn}}^2+\pi\sigma(N_pd_p^2)-E_s-\pi\sigma\sum_{i=1}^{N_{parcel}}(N_id_i^2)}{\frac{1}{2}\sum_{i=1}^{N_{parcel}}(m_iV_i^2)} \tag{12.232}$$

The threshold energy E_s is computed as:

$$E_s=\frac{1}{2}m_pV_s^2 \tag{12.233}$$

where V_s is the splashing threshold velocity computed from the critical Weber number as

$$V_s = \sqrt{We_{cr}\frac{\sigma}{\rho d_p}} \tag{12.234}$$

and the critical Weber number is given by:

$$We_{cr} = 18^2 d_p \left(\frac{\rho}{\sigma}\right)^{1/2} \left(\frac{\mu}{\rho}\right)^{1/4} \left(\frac{V_{pn}}{d_p}\right)^{3/4} \tag{12.235}$$

This correction factor is needed due to the relatively small number of sampled points for the velocity of the splashed drops (see Stanton [628] (p. 1089) for more detail). The components of the splashed parcel are multiplied by the square root of E_k in Equation 12.232 (p. 507) so that energy will be conserved. The normal and tangential velocity components of the splashed parcels are therefore given by

$$V'_{in} = \sqrt{E_K}\, V_{in} \quad \text{and} \quad V'_{it} = \sqrt{E_K}\, V_{it}$$

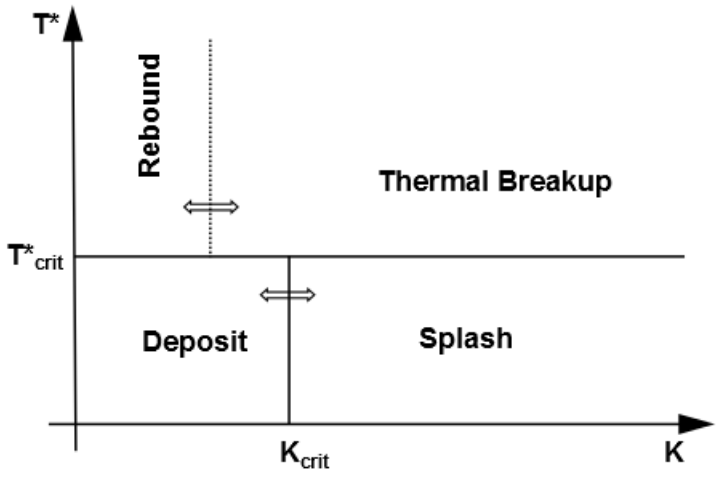
It is important to note that splashing events are inherently transient, so the splashing submodel is only available with unsteady tracking in Ansys Fluent. Splashing can also cause large increases in source terms in the cells in which it occurs, which can cause difficulty in convergence between time steps. Thus, it may be necessary to use a smaller time step during the simulation when splashing is enabled.

12.9.3.2. The Kuhnke Model

12.9.3.2.1. Regime definition

Figure 12.10: The Kuhnke Impingement Model Regimes (p. 508) shows the impingement map for the Kuhnke [322] (p. 1071) impingement model. The model considers all relevant impingement phenomena by classifying them into four regimes based on the dimensionless variables K and T^*: spread (or deposition), rebound, splash, and dry splash (thermal breakup). The deposition and splash regimes lead to formation of a wall film.

Figure 12.10: The Kuhnke Impingement Model Regimes



The dimensionless variable K is defined as:

$$K = We^{5/8} La^{1/8} \tag{12.236}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

where:

$$We = \frac{\rho V_{pn}^2 d_p}{\sigma} \tag{12.237}$$

$$La = \frac{\sigma \rho d_p}{\mu^2} \tag{12.238}$$

The dimensionless variable T^* is defined as:

$$T^* = \frac{T_w}{T_{sat}} \tag{12.239}$$

In the above equations, the following notation is used:

We = Weber number based on normal impingement velocity

La = Laplace number

ρ = droplet density (kg/m^3)

d_p = droplet diameter

V_{pn} = normal impingement velocity (m/s)

σ = droplet surface tension (N/m)

μ = droplet viscosity (kg/m-s)

T_w = wall temperature (K)

T_{sat} = droplet saturation temperature (K)

When the wall temperature T_w reaches the critical transition temperature T_{crit}, which marks the onset of film deposition, Equation 12.239 (p. 509) is used to define the critical temperature factor T^*_{crit}:

$$T^*_{crit} = T_{crit} / T_{sat} \tag{12.240}$$

More specifically, the Kuhnke model considers the following regimes:

- Spread: Adds the impinging drop to the wall film under one of the following conditions:
  - $K < K_{crit}$ and $T^* < T^*_{crit}$
  - $K < K_{crit}$ and the wall is wet
- Rebound: Emits the impinging drop without change in size under one of the following conditions:
  - $K < K_{crit}$ and $T^* > T^*_{crit}$
  - $We < 5$ and the wall is wet
- Splash: Atomizes drops into smaller ones if:
  - $K > K_{crit}$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

In the splashing regime, depending on the wall temperature and wall surface conditions, part of the impinging droplet mass may be deposited to form a film (splash), or the droplet may be completely atomized (dry splash or thermal breakup).

The critical transition parameters K_{crit} and T^*_{crit} are computed as follows.

The wall temperature-dependent transition is controlled by the critical temperature factor T^*_{crit}, which has values between 1.1 and 1.5 depending on the application. The default value for T^*_{crit} is 1; as a result, the default critical transition temperature $\mathrm{T_{crit}}$ will be equal to the saturation temperature of the droplet liquid $\mathrm{T_{sat}}$. Kuhnke [322] (p. 1071) recommends values of 1.1 for single drop impingement and 1.16 for impingement of droplet chains. For most urea-water slurries, T^*_{crit} is set at 1.4 [66] (p. 1056).

As an alternative for the Lagrangian wall film model, the critical transition temperature T_{crit} can be calculated from the saturation temperature of the liquid T_{sat} by adding a constant offset T_{cal}:

$$T_{crit}=T_{sat}+T_{cal} \tag{12.241}$$

For SCR slurries, recommended values for T_{cal} range between 167 and 207° C.

The two critical values of the parameter K (one corresponding to the transition between rebound and thermal breakup at high temperatures, K_{hot}, and the other corresponding to the transition between deposition and splash at low temperatures, K_{cold}) depend on several parameters and wall conditions (wet or dry) as further described.

- Wet wall

$$K_{crit}\left(\delta,La\right)=\varphi\left(\delta,\delta_0,125,58.7,-100\right)+\left(408.4\delta_0-283.6\right)\cdot Weibull\left(\delta,\delta_0,3\right) \tag{12.242}$$

  where the dimensionless film height δ is calculated as:

$$\delta=h_f/d_p$$

  with h_f as the film height [m] and d_p as the droplet diameter [m].

  The other parameters in Equation 12.242 (p. 510) are expressed as:

$$\varphi\left(x,x_o,y_l,y_r,s\right)=\frac{y_r-y_l}{\pi}\arctan\left(\pi s\frac{x-x_o}{y_r-y_l}\right)+\frac{y_r+y_l}{2} \tag{12.243}$$

$$\delta_0=\varphi\left(La,600,0.85,1.08,0.003\right) \tag{12.244}$$

$$Weibull\left(x,x_0,q\right)=\frac{q}{x_0}\left(\frac{x}{x_0}\right)^{q-1}e^{-\left(x/x_0\right)^q} \tag{12.245}$$

  Here x, x_o, y_l, y_r, s, and q are constants with values shown in Equation 12.242 (p. 510).

- Dry wall

$$K_{crit}\left(T^*,\varepsilon_a\right)=\left(1-\lambda\right)K_{cold}+\lambda K_{hot} \tag{12.246}$$

  where

  ε_a = dimensionless surface roughness R_a/d_p

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

R_a = surface roughness (m) (average of absolute distance from mean)

and

$$\lambda(T^*)=\arctan\left(p\left(T^*-T^*_{crit}\right)\right)/\pi+1/2 \tag{12.247}$$

with

$$p=\tan\left(\pi\left(\omega - 0.5\right)/2\right)/dT^*$$

where ω is the threshold transition value of the blending function taken as 0.99 and dT^* is the width of the blending zone.

The parameter K_{hot} is randomly sampled between 20 and 40:

$$K_{hot}=20+20p \tag{12.248}$$

where p is a uniformly distributed random number between 0 and 1:

$$p=rnd(0, 1)$$

The parameter K_{cold} is computed by the following equations:

$$K_{cold}(T^*,\varepsilon_a)=K_{\min}(T^*)-12.75\left(\ln(\varepsilon_a)-\chi\right)+12.75\sqrt{\left(\ln(\varepsilon_a)-\chi\right)^2+5} \tag{12.249}$$

with

$$\chi=-\frac{K_{\min}(T^*)+43.6}{25.5}$$

$$K_{\min}(T^*)=\begin{cases}K_{T_{\min}}+\left(K_{T_{\max}}-K_{T_{\min}}\right)e^{c_T\left(T^*-1\right)} & T^*\le 1\\ K_{T_{\max}} & 1<T^*<T^*_{trans}\end{cases} \tag{12.250}$$

where

$K_{T_{\min}}$ = 54

$K_{T_{\max}}$ = 130

c_T = 13

The splashed mass fraction is determined as:

$$y_s=\min\left[1,\ \left(T^*-0.8\right)\left(1-B\right)/\left(T^*_{crit}-0.8\right)+B\right] \tag{12.251}$$

where

$$B=\begin{cases}0.2\ +\ 0.6\ p & \text{for dry wall}\\ 0.2\ +\ 0.9\ p & \text{for wet wall}\end{cases} \tag{12.252}$$

with $p=rnd(0, 1)$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

12.9.3.2.2. Rebound

In the rebound regime, the velocity of the rebounding particle is given by:

$$V_{rt}=V_{pt} \tag{12.253}$$

$$V_{rn}=e_n V_{pn} \tag{12.254}$$

where V_{rt} and V_{rn} are the tangential and normal components of the rebounding velocity, V_{pt} is the tangential component of the impinging velocity, and e_n is the normal restitution coefficient computed from:

$$e_n=\sqrt{0.678e^{-0.04415We_0}} \tag{12.255}$$

where We_0 is the normal impact Weber number prior to impingement.

The deviation angle of the rebounding particle, ψ, takes a random value between -90° and 90° from the incident direction.

12.9.3.2.3. Splashing

Figure 12.11: The Kuhnke Model: Impinging Drop (p. 513) graphically depicts the following splashed droplet variables for the Kuhnke splashing model:

- Diameter of the impinging particle, d_p
- Impinging particle velocity, V_p plus variability
- Diameters of the splashed secondary particles, d_i
- Splashed particle mean velocity, V_i plus variability
- Impingement angle Θ_I measured from the wall surface
- Reflection angle of the splashed particle Θ_s measured from the wall surface
- Deviation angle ψ from the incident direction

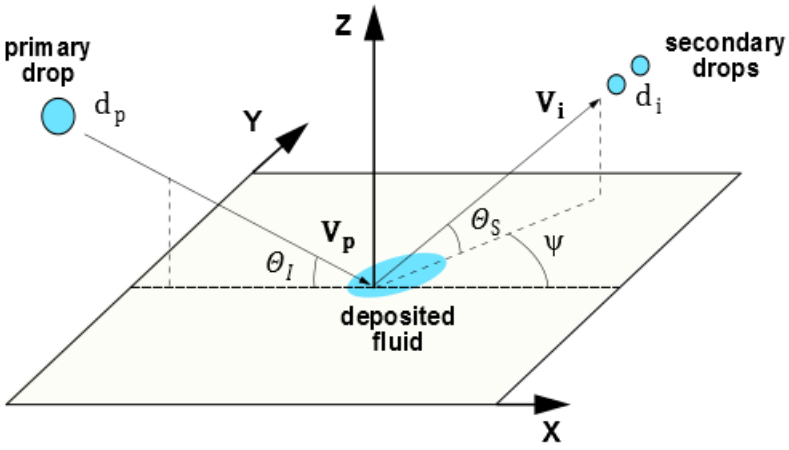
Note:

Note that, conventionally, the impingement and reflection angles are measured differently within the Stanton/Rutland and Kuhnke model frameworks:

- the Stanton and Rutland model: from the normal to the wall surface
- the Kuhnke model: from the wall surface

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Figure 12.11: The Kuhnke Model: Impinging Drop



The properties of the splashed particles are calculated as follows.

As a first step, the mean diameter of the splashed drops is computed from:

$$d_{10}=\gamma_{10}d_{\mathrm{p}} \tag{12.256}$$

with

$$\gamma_{10}=\begin{cases}3.3e^{3.6(\theta_l/\pi)^2}We^{-0.65} & \text{dry wall}\\ 2.2e^{3.6(\theta_l/\pi)^2}We^{-0.36} & \text{wet wall}\end{cases} \tag{12.257}$$

where

d_{p} = diameter of primary drop [m]

We = normal impact Weber number of the impingement particle

Θ_I = impingement angle in radians

Next, as originally proposed by Stanton and Rutland [626] (p. 1089), [628] (p. 1089), the splashed drop diameters d_{si} are randomly sampled from the inverse of a continuous Weibull Probability Density Function $weibull_{cpdfi}$ which is described by:

$$weibull_{cpdfi}\left(d_i, d_{10}, q\right) = d_{10}\left(-log\left(1-d_i\right)\right)^{1/q} \tag{12.258}$$

with q=2. A limiting method as described in Splashing (p. 503) is also applied.

The number of droplets in each splashed parcel is determined from the overall mass balance by Equation 12.222 (p. 506).

The mean splashed droplet velocity V_1 is calculated from the secondary Weber number We_{a1} and mean droplet size d_{10} as:

$$V_1=\sqrt{\frac{\sigma We_{a1}}{\rho d_{10}}} \tag{12.259}$$

where

$$We_{a1}=\begin{cases}\gamma_{10}\left(We_{a0}\left(1-\eta\sin^2\Theta_I\right)+12\right)-12/\nu_{32} & \text{dry wall}\\ \max\left(51-7.1e^{3.4(\Theta_I/\pi)},\ c_aWe_{a0}\right) & \text{wet wall}\end{cases} \tag{12.260}$$

where We_{a0} and We_{a1} are based on absolute particle velocities, and

$\eta = 0.85$

$c_a=-0.378\left(\Theta_I/\pi\right)^2-0.123\left(\Theta_I/\pi\right)+0.156$

$\nu_{32} = 2$

Similarly to the splashed droplet diameters, the splashed drop velocities V_i are randomly sampled from the inverse of a continuous Weibull Probability Density Function with the distribution average value being V_1.

The ejection angle Θ_S is computed from the mean value $\overline{\Theta}_S$ by assuming a logistic distribution:

$$\overline{\Theta}_S=\begin{cases}9.3+\ln\left(\varepsilon_z\right)\left(2.7-0.03\Theta_I\right)+0.22\Theta_I & \text{cold dry wall}\\ 0.225\Theta_I e^{\left(0.017\Theta_I-0.937\right)^2} & \text{wet wall}\\ 0.96\Theta_I e^{-0.0045We} & \text{hot wall}\end{cases} \tag{12.261}$$

where angles (in degrees) are distributed by the logistic PDF $logistic\left(\Theta_S,\ \overline{\Theta}_S,\ \sigma_l\right)$ with σ_l=4.

The notation used here is as follows:

ε_z = dimensionless surface roughness R_z/d_p

R_z = surface roughness (m) (average of maximal height differences)

The inverse of the continuous logistic PDF is given by:

$$\log istic_{cpdfi}\left(\Theta_S,\ \overline{\Theta}_S,\ \sigma_l\right)=\sigma_l\log\left[\Theta_S/\left(1-\Theta_S\right)\right]+\ \overline{\Theta}_S \tag{12.262}$$

The deviation angle ψ is determined by:

$$\psi=-\frac{\pi}{\beta}\ln\left(1-p\left(1-e^{-\beta}\right)\right),\ p=rnd\left(0,\ 1\right) \tag{12.263}$$

where

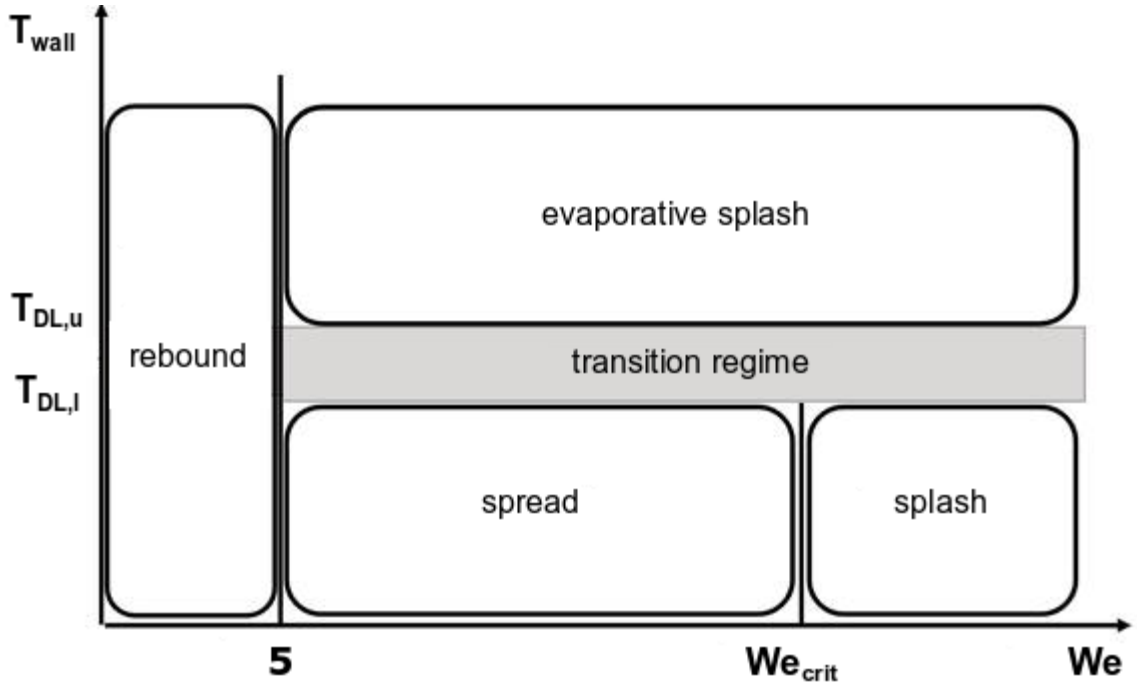
$$\beta=\begin{cases}\sqrt{\left(1+8.872\cos\left(1.152\Theta_I\right)\right)/\left(1-\cos\Theta_I\right)} & \Theta_I\le 80^\circ\\ \left(\pi^2/2\right)\cos\Theta_I & \Theta_I\ge 80^\circ\end{cases}$$

12.9.3.3. The Stochastic Kuhnke Model

The regime map of the Stochastic Kuhnke model is shown in Figure 12.12: The Regime Map of the Stochastic Kuhnke Model (p. 515).

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Figure 12.12: The Regime Map of the Stochastic Kuhnke Model



The model considers all relevant impingement phenomena by classifying them into the following four core regimes based on critical temperature and Weber number transition criteria:

- rebound
- spread
- splash
- evaporative splash

The spread and splash regimes lead to formation of a wall film. In addition, the transition regime is introduced to avoid abrupt changes during the film deposition process.

The regime transition criteria are defined as follows. A critical temperature range is defined between the upper and lower deposition limits $T_{DL,u}$ and $T_{DL,l}$:

$$T_{DL,u} = T_{sat} + \Delta T_{DL} \tag{12.264}$$

$$T_{DL,l} = T_{DL,u} - \Delta T_{lim} \tag{12.265}$$

where

T_{sat} = liquid film saturation temperature [K]

ΔT_{DL} = upper deposition limit offset [K]

ΔT_{lim} = deposition range temperature difference [K]

The Weber number criterion is defined differently for wet and dry walls:

- Critical Weber number for splashing on a wet wall:

$$We_{crit,w} = A_w La^{-0.18} \tag{12.266}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

- Critical Weber number for splashing on a dry wall:

$$We_{crit,D} = A \cdot La^{-0.18} \tag{12.267}$$

In Equation 12.266 (p. 515) and Equation 12.267 (p. 516):

La = Laplace number

A_w = Laplace number constant

A = function of wall roughness r_z interpolated from Table 12.2: Parameter *A* as a Function of Wall Roughness (p. 516)

Table 12.2: Parameter *A* as a Function of Wall Roughness

r_z (microns)	A
0.05	5264
0.14	4534
0.84	2634
3.1	2056
12	1322

The Laplace number is expressed as:

$$La = \rho\sigma d / \mu^2 \tag{12.268}$$

where

ρ = droplet density

σ = droplet surface tension

d = droplet diameter

μ = droplet viscosity

Finally, the Stochastic Kuhnke model assigns the appropriate regime for each impingement event according to the droplet Weber number (see the definition of the Weber number in section "The Kuhnke Model" in the Ansys Fluent Theory Guide) and the wall conditions. These regimes are:

- Rebound

  Regime boundaries: $We < 5$ and $\begin{cases} T_{wall} < T_{DL,l} \text{ and Wet wall} \\ or \\ T_{wall} > T_{DL,l} \end{cases}$

  The impinging drop is emitted without change in size. The velocity components of the rebounding particle are calculated in accordance with section "Rebound" for the Kuhnke model in the Ansys Fluent Theory Guide.

- Spread

  Regime boundaries: $T_{wall} < T_{DL,l}$ and $\begin{cases} 5 < \mathrm{We} < \mathrm{We_{SW}} \text{ and Wet wall} \\ or \\ \mathrm{We} < \mathrm{We_{SD}} \text{ and Dry wall} \end{cases}$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

The impinging drop is added to the wall film.

- Splash

  Regime boundaries: $T_{wall} < T_{DL,l}$ and $\begin{cases} \mathrm{We} > \mathrm{We}_{\mathrm{SW}} \text{ and Wet wall} \\ or \\ \mathrm{We} > \mathrm{We}_{\mathrm{SD}} \text{ and Dry wall} \end{cases}$

  Part of the impinging droplet mass is deposited to form a film, and the rest of the droplet is atomized. The splashed mass fraction and the splashed droplet variables are computed according to section "Splashing" for the Kuhnke model in the Ansys Fluent Theory Guide.

- Evaporative Splash

  Regime boundaries: $T_{wall} > T_{DL,u}$ and $We > 5$

  A small fraction of the impinging liquid spray is assumed to vaporize immediately upon impact, while the rest is completely atomized emitting smaller sized droplets. The vaporization events occurring in this regime are controlled by random number sampling. Similarly to the splash regime, the splashed droplet variables are computed as described in section "Splashing" for the Kuhnke model in the Ansys Fluent Theory Guide.

- Transition Regime

  Regime boundaries: $T_{DL,l} < T_{wall} < T_{DL,u}$ and $We > 5$

  Under these conditions, any of the regimes (spread, splash, or evaporative splash) may be assigned to the impinging drop. This is decided based on comparison of a random number with the deposition ratio dr computed from Equation 12.269 (p. 517) in combination with the droplet Weber number and the wet or dry wall criterion.

$$dr = \frac{T_{wall} - T_{DL,l}}{T_{DL,u} - T_{DL,l}} \tag{12.269}$$

12.9.4. Separation and Stripping Submodels

For a detailed description of the film separation models, see Film Separation (p. 842). For details about the film stripping model, see Film Stripping (p. 844).

12.9.5. Conservation Equations for Wall-Film Particles

Conservation equations for momentum, mass, and energy for individual parcels in the wall-film are described below. The particle-based approach for thin films was first formulated by O'Rourke [492] (p. 1081) and most of the following derivation is based closely on that work.

12.9.5.1. Momentum

The equation for the momentum of a parcel in the film is:

$$\rho h \frac{d\vec{u_p}}{dt} = \tau_g \vec{t} + \vec{\tau_w} + \vec{F} + \rho h \left(\vec{g} - \vec{\alpha_w} \right) \tag{12.270}$$

where

$\vec{u_p}$ = is the film particle velocity

τ_g = is the magnitude of the shear stress of the gas flow on the surface of the film

$\vec{t}$ = is the unit vector in the direction of the film surface velocity

$\vec{\tau_w}$ = is the stress that the wall exerts on the film

$\vec{F}$ = is the force per unit area necessary to keep the film on the surface

h = is the current film height at the particle location

$\rho h\left(\vec{g}-\vec{\alpha_w}\right)$ = is the body force term

Despite the small values of film thickness, the body force term can be very significant due to the very high acceleration rates seen in simulations with moving boundaries.

The expression for the wall stress, $\vec{\tau_w}$, is:

$$\vec{\tau_w}=-2\frac{\mu_l}{h}\left(\vec{u_p}-\vec{u_w}\right) \tag{12.271}$$

where μ_l is the liquid film viscosity and $\vec{u_w}$ is the wall velocity.

The force $\vec{F}$ is included in Equation 12.270 (p. 517) for completeness, but is not modeled in the formulation of the equations solved in Fluent. Instead, it is accounted for by explicitly enforcing the condition:

$$\vec{u_p}\cdot\hat{n}_\alpha=0 \tag{12.272}$$

at each time step for all particles representing the wall-film.

Ansys Fluent solves a particle position equation of the form:

$$\frac{d\vec{u_p}}{dt}=\alpha-\beta\vec{u_p} \tag{12.273}$$

After rearranging Equation 12.270 (p. 517) and substituting Equation 12.271 (p. 518) the film particle acceleration is given by:

$$\frac{d\vec{u_p}}{dt}=\frac{\tau_g\vec{t}}{\rho h}+\frac{2\mu_l\vec{u_w}}{\rho h^2}+\left(\vec{g}-\vec{\alpha_w}\right)-\frac{2\mu_l}{\rho h^2}\vec{u_p} \tag{12.274}$$

12.9.5.2. Mass Transfer from the Film

12.9.5.2.1. Film Vaporization and Boiling

The film vaporization law is applied when the film particle temperature is above the vaporization temperature T_{vap}. In Ansys Fluent, the vaporization rate is modeled using following models:

- Diffusion Controlled Model
- Convection/Diffusion Controlled Model
- Gas-Side Boundary Layer Model

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

- Film Boiling Model

12.9.5.2.1.1. Diffusion Controlled Model

The film vaporization rate is governed by gradient diffusion from the surface exposed to the gas phase. The gradient of vapor concentration between the film surface and the gas phase is

$$\dot{N}_i = k_{film}(C_{i,s} - C_{i,\infty}) \tag{12.275}$$

where $\dot{N}_i$ is the molar flux of vapor (kmol/m^2-s), k_{film} is the mass transfer coefficient (m/s), and $C_{i,s}$ and $C_{i,\infty}$ are the vapor concentrations on the film surface and in the bulk gas, respectively (kmol/m^3).

The vapor concentration at the film surface is evaluated using the saturated vapor pressure at the film temperature, and the bulk gas concentration is obtained from the flow field solution. The vaporization rate is sensitive to the saturated vapor pressure, similar to the droplet vaporization.

The film mass transfer coefficient is obtained as:

$$k_{film} = Nu \cdot D_i / len \tag{12.276}$$

where D_i is the binary diffusivity of species i, Nu is the Nusselt correlation, and the len is the characteristic length (in meters) computed as the square root of the film parcel area A_p:

$$len = \sqrt{A_p} \tag{12.277}$$

A_p is taken to be a mass weighted percentage of the wall face area A_w:

$$A_p = A_w \cdot m_p / m_{film} \tag{12.278}$$

In Equation 12.276 (p. 519) for the mass transfer coefficient, the Nusselt correlation is used after replacing the Prandtl number with the Schmidt number as follows:

- Laminar Flow:

$$Nu = 0.332\ Re^{1/2} Sc^{1/3} \tag{12.279}$$

- Turbulent Flow:

$$Nu = 0.0296\ Re^{4/5} Sc^{1/3} \tag{12.280}$$

where the Reynolds number Re is calculated from the particle film characteristic length len and the particle relative velocity component that is parallel to the wall.

For multicomponent vaporization, the diffusivity of each species is used in Equation 12.276 (p. 519) and Equation 12.279 (p. 519) to obtain the vaporization rate of each component in the liquid film mixture.

The mass of the particle parcel m_p is decreased according to:

$$m_p(t + \Delta t) = m_p(t) - N_i A_p M_{w,i} \Delta t \tag{12.281}$$

where $M_{w,i}$ is the molecular weight of the gas phase species i to which the vapor from the liquid is added. The diameter of the film particle is decreased to account for the mass loss in the individual parcel. This keeps the number of drops in the parcel constant.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

12.9.5.2.1.2. Convection/Diffusion Controlled Model

The film vaporization rate is governed by:

$$\dot{N}_i = k_{film}\,\rho_{\infty}\ln(1+B_m) \tag{12.282}$$

$$B_m = \frac{Y_{i,surf} - Y_{i,\infty}}{1 - Y_{i,surf}}$$

where

$\dot{N}_i$ = vapor mass flux of species i (kg/m^2/s)

$Y_{i,surf}$ and $Y_{i,\infty}$ = mass fraction of species i on the film surface and the bulk respectively

k_{film} = film mass transfer coefficient (m/s) calculated from Equation 12.276 (p. 519)

ρ_{∞} = is density of the bulk gas (kg/m^3)

12.9.5.2.1.3. Thermolysis Model

The single-rate thermolysis model uses Equation 12.87 (p. 468) and the constant rate model uses Equation 12.88 (p. 468). If the Lagrangian wall film is considered, the secondary rate model that allows you to differentiate between the thermolysis rates of free-stream particles and film particles can be applied. The model uses Equation 12.87 (p. 468) to calculate separately the mass transfer rate from the free particle to the bulk gas phase (using the particle pre-exponential factor $A_{particle}$ and the particle activation energy $E_{particle}$) and the mass transfer rate from the film to the bulk gas phase (using the film pre-exponential factor A_{film} and the film activation energy E_{film}).

12.9.5.2.1.4. Gas-Side Boundary Layer Model

When the partial pressure of species in the bulk is less than the vapor pressure at the film surface, Fluent uses Equation 12.291 (p. 523) and Equation 12.292 (p. 523) to calculate the vaporization rate for laminar and turbulent flows, respectively. The equations are using wall functions to determine the mass transfer coefficients. See Film Condensation (p. 522) for details.

12.9.5.2.1.5. Film Boiling Model with Conduction Heat Transfer Model

When the film temperature reaches the boiling point, the film boiling rate is calculated by setting the right-hand side of the energy balance equation (Equation 12.301 (p. 526)) to 0 in and solving for the film boiling rate $\dot{m}_p$.

The resulting boiling rate expressions are as follows:

Constant Temperature Walls

$$\dot{m}_p = \frac{A_p}{h_{fg}}\left[-k_p\left(1+\frac{h_{film}}{k_p+h_{film}}\right)T_p + k_p T_w + h_{film}\left(1-\frac{h_{film}}{k_p+h_{film}}\right)T_{\infty}\right] \tag{12.283}$$

Heat Flux Walls

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$\dot{m}_p = \frac{A_p}{h_{fg}} \left[-\left(\frac{k_p h_{film}}{k_p + h_{film}} \right) T_p + \dot{q}_w + h_{film} \left(1 - \frac{h_{film}}{k_p + h_{film}} \right) T_\infty \right] \tag{12.284}$$

with:

$$k_p = k_l / (h/2) \tag{12.285}$$

$$h_{film} = \frac{k_\infty}{len} Nu$$

where

$\dot{m}_p$ = film parcel boiling rate (kg/s)

A_p = the film parcel area (m^2)

h_{fg} = latent heat (J/kg)

T_p = film parcel temperature (K)

T_w = wall temperature (K)

T_∞ = temperature in the bulk phase (K)

$\dot{q}_w$ = wall heat flux (W/m^2)

h = the film height (m)

k_l = thermal conductivity of the liquid film (W/m/K)

If the Gas-Side Boundary Layer model is enabled, the heat transfer coefficient h_{film} is computed as follows.

- Laminar flow: $h_{film} = k_\infty / l$, where the length l is the normal distance from film surface to the bulk.
- Turbulent flow: h_{film} is determined from the wall functions as described in Energy Transfer from the Film (p. 524).

In other cases, Fluent uses the Nusselt number expression:

$$h_{film} = k_\infty Nu / len \tag{12.286}$$

where

k_∞ = thermal conductivity of the continuous phase (W/m/K)

len = the characteristic length (m)

The Nusselt number Nu is computed from Equation 12.299 (p. 526). If the film is cooled below the boiling point by contacting a cold wall, the vaporization rate calculation will revert to the vaporization equations (Equation 12.275 (p. 519) or Equation 12.282 (p. 520)).

12.9.5.2.1.6. Film Boiling Model with Convective Heat Transfer Model

The boiling rate expressions for the convective heat transfer model are as follows:

For constant temperature walls:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$\frac{dm_p}{dt} = \left[-\frac{1}{1-f(\mathrm{Pr})}\left(h_{wall}+h_\infty\right)T_p + h_{wall}T_w + f(\mathrm{Pr})\left(\frac{h_{wall}+h_\infty}{1-f(\mathrm{Pr})}\right)T_w + h_\infty T_\infty\right]\frac{A_p}{h_{lat}}$$
$$= \left[-\frac{1}{1-f(\mathrm{Pr})}\left(h_{wall}+h_\infty\right)T_p + \left(\frac{h_{wall}+f(\mathrm{Pr})h_\infty}{1-f(\mathrm{Pr})}\right)T_w + h_\infty T_\infty\right]\frac{A_p}{h_{lat}} \quad (12.287)$$

For heat flux walls:

$$\frac{dm_p}{dt} = \left[-h_\infty T_p + \frac{Q_{wall}}{A_p}\left(1+f(\mathrm{Pr})\frac{h_\infty}{h_{wall}}\right) + h_\infty T_\infty\right]\frac{A_p}{h_{lat}} \quad (12.288)$$

with

$$f(\mathrm{Pr}) = \frac{3}{8}\mathrm{Pr}^{-1/3} \quad (12.289)$$

In the above equations,

m_p = mass of the film particle (kg)

T_p = film particle temperature (K)

T_∞ = bulk temperature (K)

T_w = wall temperature (K)

h_{lat} = latent heat of vaporization (J/kg)

h_{wall} = heat transfer coefficient from wall to liquid film (W/K-m^2)

h_∞ = heat transfer coefficient from liquid film to bulk phase (W/K-m^2)

A_p = film parcel area, taken to be a mass weighted percentage of the wall face area (m^2)

Q_{wall} = heat flux imposed on the wall (W)

Pr = liquid Prandtl number $\mathrm{Pr} = \frac{c_{p,l}\mu_l}{k_l}$

The heat transfer coefficient from liquid film to bulk phase (h_∞) is obtained as described in Film Boiling Model with Conduction Heat Transfer Model (p. 520).

The heat transfer coefficient from wall to liquid film h_{wall} is obtained from Equation 12.312 (p. 529).

If the film is cooled below the boiling point by contacting a cold wall, the vaporization rate calculation will revert to the vaporization equations (Equation 12.275 (p. 519) or Equation 12.282 (p. 520)).

12.9.5.2.2. Film Condensation

The condensation model is triggered when the partial pressure of species i in the bulk exceeds its vapor pressure at the film surface. The Fluent Lagrangian wall film model uses two different expressions for wall film condensation under laminar and turbulent flow conditions. The derivation follows [736] (p. 1095), assuming that the condensation of species i from the gas phase takes place in the presence of noncondensables.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

In the Ansys Fluent Lagrangian Wall Film Condensation model, component i condenses on the wall film in the presence of non-condensable components. For the condensation rate of component i through a plane at a distance l from the film surface, Fick's law can be written as:

$$\dot{m} = -D_i \rho_\infty \left| \left(1 - y_i\right) \frac{\partial y_i}{\partial l} \right. \tag{12.290}$$

where

$\dot{m}$ = mass flux of vapor of species (kg/m^2-s)

ρ_∞ = density of the bulk phase (kg/m^3)

D_i = diffusion coefficient of species i (m^2/s)

y_i = mass fraction of species i

For laminar flow, the mass fraction of species i is integrated from the film surface to the bulk gas conditions yielding the logarithmic term, while in case of turbulent flow, finite differences are taken directly. The final expressions are:

- Laminar boundary layer:

$$\dot{m} = k_{film} \rho_\infty ln\left[\frac{1 - Y_{i\infty}}{1 - Y_{is}}\right] \tag{12.291}$$

- Turbulent boundary layer:

$$\dot{m} = -k_{film} \rho_\infty \frac{Y_{i\infty} - Y_{is}}{1 - Y_{is}} \tag{12.292}$$

where

k_{film} = mass transfer coefficient (m/s)

Y_{is} and $Y_{i\infty}$ = mass fraction of species i at the film surface and the bulk, respectively

For laminar flow, the mass transfer coefficient k_{film} in Equation 12.291 (p. 523) is computed as:

$$k_{film} = \frac{D_i}{l} \tag{12.293}$$

where l is the normal distance from the film surface to the bulk. l is computed as the normal distance from the wall adjacent cell center to the wall.

For turbulent flow, the turbulent mass transfer coefficient k_{film} in Equation 12.292 (p. 523) is determined from the wall functions. The current implementation uses wall functions as described by Equation 4.348 (p. 141). When the enhanced wall treatment is enabled in the turbulence model, the implementation follows the calculations in Enhanced Wall Treatment ε-Equation (EWT-ε) (p. 145).

For multicomponent particles, Equation 12.291 (p. 523) or Equation 12.292 (p. 523) are applied to each liquid component.

See Film Condensation Model in the *Fluent User's Guide* for more information on how to use the film condensation model.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

12.9.5.2.3. Multicomponent Film Models

For the multicomponent Lagrangian wall film, the vaporization rate is calculated as the sum of the vaporization rates of the individual components. The vaporization rate of each component in the liquid film are computed from:

- Diffusion Controlled model: Equation 12.275 (p. 519) and Equation 12.276 (p. 519)
- Convection/Diffusion Controlled model: Equation 12.282 (p. 520)

If the Thermolysis model has been enabled for any component in the film, Ansys Fluent applies Equation 12.164 (p. 486) to obtain the thermolysis rate of the individual component for the single rate or Equation 12.165 (p. 486) to obtain the constant rate.

If the Lagrangian wall film is considered, the secondary rate model can be used. This model uses Equation 12.164 (p. 486) to calculate separately the vaporization rate of the free-stream particle (using the particle pre-exponential factor $A_{particle}$ and the particle activation energy $E_{particle}$) and the vaporization rate of the film (using the film pre-exponential factor A_{film} and the film activation energy E_{film}).

When the total vapor pressure at the multicomponent film surface exceeds the cell pressure, Ansys Fluent applies a multicomponent film boiling equation by using averaged forms of Equation 12.283 (p. 520) or Equation 12.284 (p. 521) in accordance with the thermal boundary condition applied at the wall. The average latent heat h_{fg} is computed according to Equation 12.170 (p. 487) from the latent heats of the individual liquid components and the overall boiling rate of the film is calculated by averaging the rates computed from Equation 12.283 (p. 520) or Equation 12.284 (p. 521) for the conduction heat transfer model, and from Equation 12.287 (p. 522) or Equation 12.288 (p. 522) for the convective heat transfer model, for each liquid component for each liquid component with the fractional vaporization computed by Equation 12.171 (p. 487).

12.9.5.3. Energy Transfer from the Film

12.9.5.3.1. Conduction Heat Transfer Model

To obtain an equation for the temperature in the film, energy flux from the gas side as well as energy flux from the wall side must be considered. In the absence of film evaporation/boiling, the energy transfer from/to the film particle is then given by:

$$\frac{d}{dt}\left(m_p c_p T_p\right) = Q_{p,cond} + Q_{p,conv} \tag{12.294}$$

where

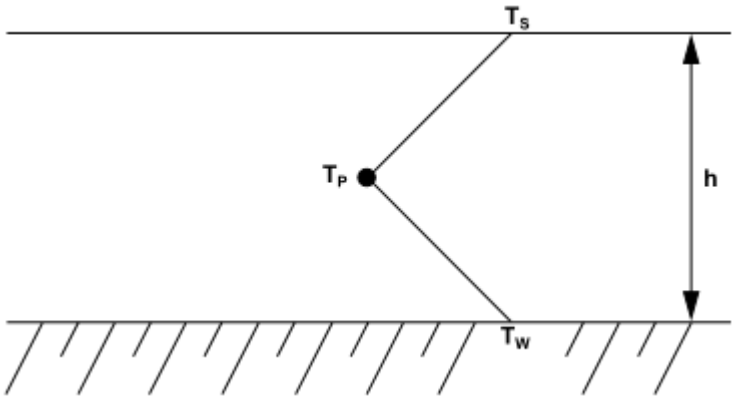
m_p, c_p, and T_p = mass, specific heat and temperature of the film parcel, respectively

$Q_{p,cond}$ = the conduction from the wall

$Q_{p,conv}$ = the convective heat flux at the film surface

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Figure 12.13: Assumption of a Bilinear Temperature Profile in the Film



When the thermal boundary condition on the wall is set to a constant temperature, $Q_{p,cond}$ is calculated as:

$$Q_{p,cond} = \frac{k_l A_p}{(h/2)}\left(T_w - T_p\right) \tag{12.295}$$

where

k_l = thermal conductivity of the liquid film

A_p = the film parcel area computed from Equation 12.278 (p. 519)

h = the film height

For walls where the heat flux, $\dot{q}_w$ is imposed by the boundary condition, Equation 12.295 (p. 525) is replaced by:

$$Q_{p,cond} = \dot{q}_w A_p \tag{12.296}$$

The convective heat flux, $Q_{p,conv}$, is calculated as:

$$Q_{p,conv} = A_p h_{film}\left(T_\infty - T_s\right) \tag{12.297}$$

where

h_{film} = heat transfer coefficient (W/K-m^2)

T_s = film surface temperature

If the film condensation or the gas-side boundary layer model is enabled, the heat transfer coefficient h_{film} is computed as follows.

- Laminar flow:

$$h_{film} = \frac{k_\infty}{l} \tag{12.298}$$

  where l is the normal distance from the film surface to the bulk. l is computed as the normal distance from the wall adjacent cell center to the wall.

- Turbulent flow:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

h_{film} is determined from the wall functions. The current implementation uses wall functions as de- scribed in Standard Wall Functions (p. 138), Equation 4.343 (p. 140). When the enhanced wall treatment is enabled in the turbulence model, the implementation follows the calculations in Enhanced Wall Treatment ε-Equation (EWT-ε) (p. 145)

In other cases, Fluent uses a Nusselt number expression for h_{film}:

$$h_{film} = k_{\infty} Nu / len$$

where

k_{∞} = thermal conductivity of the continuous phase (W/m/K)

len = the characteristic length (m) computed from Equation 12.277 (p. 519)

The Nusselt number Nu is calculated as:

$$\begin{aligned} &\text{Laminar Flow:} \quad Nu = 0.332\ Re^{1/2} Pr^{1/3} \\ &\text{Turbulent Flow:} \quad Nu = 0.0296\ Re^{4/5} Pr^{1/3} \end{aligned} \tag{12.299}$$

The film surface temperature T_s in Equation 12.297 (p. 525) is determined from the interface conservation relation:

$$h_{film}\left(T_{\infty} - T_s\right) = \frac{k_l}{\left(h/2\right)}\left(T_s - T_p\right) \tag{12.300}$$

When the particle changes its mass during vaporization, an additional term is added to Equation 12.294 (p. 524) to account for the enthalpy of vaporization:

$$\frac{d}{dt}\left(m_p c_p T_p\right) = Q_{p,cond} + Q_{p,conv} - \dot{m}_p\ h_{fg} \tag{12.301}$$

where $\dot{m}_p$ is the film parcel vaporization rate (kg/s), and h_{fg} is the latent heat of vaporization (J/kg).

As the film parcel trajectory is computed, Ansys Fluent integrates Equation 12.294 (p. 524) or Equation 12.301 (p. 526) and obtains the temperature at the next time step.

The implementation does not account for film radiation effects. The energy source terms are calculated for the film parcels from the energy difference along the film trajectory. For the cases where the film resides on a wall with heat-flux imposed, the heat losses or gains from the film to the wall equal the heat-flux values. For all other thermal boundary conditions, except heat flux, Fluent determines the film temperature based on the wall temperature and the heat transferred from the bulk fluid to the film. The total heat flux to the wall is then calculated by adding the heat flux from the bulk fluid to the wall and the heat flux from the film to the wall.

12.9.5.3.2. Convective Heat Transfer Model

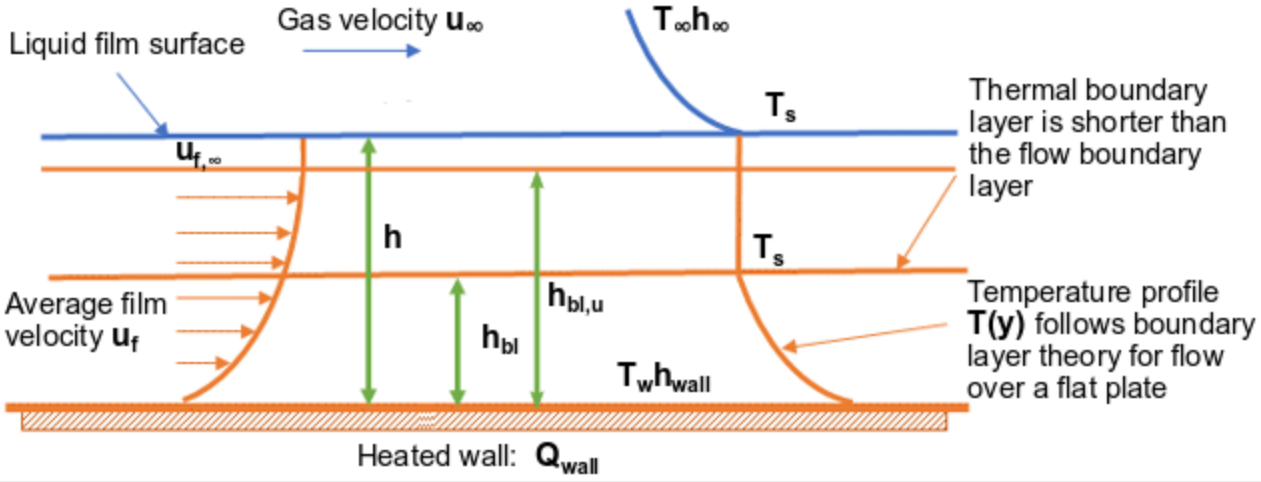
The convective wall-film heat transfer model provides a more accurate calculation of heat transfer between the Lagrangian wall film and the wall surface where it is formed. It is especially useful for thicker wall films, films moving at high velocity and where the film liquid material has a high Prandtl number. Refer to Setting the Lagrangian Wall Film Model in the *Fluent User's Guide* to learn how to use this model.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

The following assumptions are made for the flow and heat transfer in the liquid film:

- The film flow is laminar.
- The velocity and temperature profiles in the liquid film follow the laminar boundary-layer theory for flow over a flat plate as described in [369] (p. 1074).
- The velocity boundary layer $h_{bl,u}$ is located above the thermal boundary layer h_{bl}.
- For film heights significantly lower than the thermal boundary layer, the heat transfer reverts to pure conduction.
- The film is always in complete contact with the wall.

Figure 12.14: Convective Wall-Film Heat Transfer: Film Flow Over a Heated Surface



The energy balance of a film particle at temperature T_p exchanging heat with a wall surface and a bulk phase can be expressed as:

$$\frac{d}{dt}\left(m_p c_{p,l} T_p\right) = Q_{wall} + Q_{conv} + \frac{dm_p}{dt} h_{lat} \tag{12.302}$$

with

$$Q_{conv} = A_p h_\infty \left(T_\infty - T_s\right) \tag{12.303}$$

and

$$Q_{wall} = \begin{cases} A_p h_{wall}\left(T_w - T_s\right) & \text{for a constant temperature wall} \\ Q_{ext} A_p / A_{wall} & \text{for a wall with constant heat flux} \end{cases} \tag{12.304}$$

where

m_p = mass of the film particle (kg)

$c_{p,l}$ = heat capacity of the liquid (J/K-kg)

T_p = film particle temperature (K)

h_{lat} = latent heat of vaporization (J/kg)

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

h_{wall} = heat transfer coefficient from wall to liquid film (W/K-m^2)

h_∞ = heat transfer coefficient from liquid film to bulk phase (W/K-m^2)

A_{wall} = wall face area (m^2)

A_p = film parcel area, taken to be a mass weighted percentage of the wall face area A_{wall} (m^2)

T_∞ = bulk temperature (K)

T_s = film surface temperature (K)

T_w = wall temperature (K)

Q_{ext} = heat flux imposed on the wall (W)

The temperature profile inside the thermal boundary layer in the direction y normal to the wall is expressed as:

$$\frac{T-T_s}{T_w-T_s}=1-\frac{3}{2}\frac{y}{h_{bl}}+\frac{1}{2}\left(\frac{y}{h_{bl}}\right)^3 \tag{12.305}$$

where h_{bl} is the boundary layer height.

The film temperature above the thermal boundary remains constant at T_s.

The average temperature inside the boundary layer $T_{p,bl}$ can be obtained by integrating along the y direction:

$$T_{p,bl}=\overline{T}=\frac{1}{h_{bl}}\int_0^{h_{bl}}\left[T_s+\left(T_w-T_s\right)\left(1-\frac{3}{2}\frac{y}{h_{bl}}+\frac{1}{2}\left(\frac{y}{h_{bl}}\right)^3\right]dy \tag{12.306}$$

which results in:

$$T_{p,bl}=\frac{3}{8}T_w+\frac{5}{8}T_s \tag{12.307}$$

The average film parcel temperature T_p can be obtained by weighting $T_{p,bl}$ and T_s with the heights h_{bl} and ($h-h_{bl}$) under the assumptions that the film velocity and density remain constant:

$$T_p=\frac{h_{bl}}{h}T_{p,bl}+\frac{\left(h-h_{bl}\right)}{h}T_s \tag{12.308}$$

where h is the liquid film height.

For Pr > 0.6, the ratio of the thermal boundary layer thickness to the velocity boundary layer thickness can be represented as:

$$\frac{h_{bl}}{h}=Pr^{-1/3} \tag{12.309}$$

The surface temperature T_s can be calculated by combining equations Equation 12.307 (p. 528) - Equation 12.309 (p. 528) for the constant temperature walls:

$$T_s=\frac{T_p-\frac{3}{8}Pr^{-1/3}T_w}{1-\frac{3}{8}Pr^{-1/3}} \tag{12.310}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

For walls where the constant heat flux Q_{ext} is imposed, the corresponding equation for the surface temperature is expressed as:

$$T_s = T_p - \frac{\frac{3}{8} Pr^{-1/3} Q_{wall}}{A_p h_{wall}} \tag{12.311}$$

As the film parcel trajectory is computed, Ansys Fluent integrates Equation 12.302 (p. 527) and obtains the temperature at the next time step.

The model implementation does not account for film radiation effects. The energy source terms are calculated for the film parcels from the energy difference along the film trajectory. For cases where the film resides on a wall with heat-flux imposed, the heat losses or gains from the film to the wall equal the heat-flux values. For all other thermal boundary conditions, except heat flux, Ansys Fluent determines the film temperature based on the wall temperature and the heat transferred from the bulk fluid to the film. The total heat flux to the wall is then calculated from Equation 12.304 (p. 527) .

Calculation of the heat transfer coefficient h_{wall}

The heat transfer coefficient from film to wall is expressed as:

$$h_{wall} = \frac{k_l Nu}{len} \tag{12.312}$$

where k_l is the thermal conductivity of the liquid film, and len is the characteristic length.

The Nusselt number Nu is calculated from the laminar boundary-layer theory for $Pr_l > 0.6$:

- Constant temperature walls:

$$Nu = 0.332 \; Re^{1/2} Pr_l^{1/3} \tag{12.313}$$

- Constant heat flux walls:

$$Nu = 0.4587 \; Re^{1/2} Pr_l^{1/3} \tag{12.314}$$

In the above equations, the Reynolds number Re and the Prandtl number Pr_l are calculated as:

$$Re = \frac{\rho_l u_{f,\infty} len}{\mu_l} \tag{12.315}$$

$$Pr_l = \frac{c_{p,l} \mu_l}{k_l} \tag{12.316}$$

where

ρ_l, μ_l, k_l, and $c_{p,l}$ = density, viscosity, thermal conductivity and heat capacity of the liquid film, respectively.

$u_{f,\infty}$ = velocity of the liquid film in the bulk, that is, above the film flow boundary layer (m/s)

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Calculation of $u_{f,\infty}$

The velocity profile for fully developed flow over a flat plate in the laminar boundary layer is described in the direction y normal to the wall by:

$$u = u_{f,\infty}\left[\frac{3}{2}\frac{y}{h_{bl,u}} - \frac{1}{2}\left(\frac{y}{h_{bl,u}}\right)^3\right] \tag{12.317}$$

The flow boundary layer height $h_{bl,u}$ can be obtained as function of distance x from the edge of the flat plate [369] (p. 1074):

$$\frac{h_{bl,u}}{x} = \frac{4.64}{\sqrt{Re_x}} \tag{12.318}$$

The bulk velocity $u_{f,\infty}$ is calculated as a function of the average wall film velocity u_f for the following regimes:

- For $h > 2/3\, h_{bl,u}$:

$$u_{f,\infty} = 8/5\, u_f \tag{12.319}$$

- For $h < 2/3\, h_{bl,u}$, $u_{f,\infty}$ is obtained by extrapolating the cubic velocity profile from h to $h_{bl,u}$.

Considerations on the characteristic length len

len is the distance of the current particle location from the impingement point. The impingement location is known for new film particles only and can only be recorded during the impingement event. It cannot be back-calculated when reading film particles from case/data files created in earlier versions. For those old particles, the film height is used as the characteristic length len together with the below expressions for the Nusselt number Nu providing the averaged heat transfer coefficients over a specified length.

The Nusselt number for $Pr_l > 0.6$ is expressed as:

- Constant temperature walls:

$$Nu = 0.664\ Re^{1/2} Pr_l^{1/3} \tag{12.320}$$

- Constant heat flux:

$$Nu = 0.688\ Re^{1/2} Pr_l^{1/3} \tag{12.321}$$

Considerations on the heat transfer coefficient

The heat transfer coefficient h_{wall} is computed by Equation 12.312 (p. 529) as a function of the Nusselt number. However, it reverts to the conduction heat transfer coefficient $h_{wall,cond}$ under the below conditions:

- The Reynolds number is small, which may happen if the film velocity is low, or at the beginning of the film particle path

- The film height is located well below the thermal boundary layer $h < 2/3\, h_{bl}$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

In those cases, the temperature profile in the film is assumed to be linear, and the characteristic length will be equal to the film height $len=h$.

The conduction heat transfer coefficient is computed as:

$$h_{wall,cond}=\frac{k_l}{h} \tag{12.322}$$

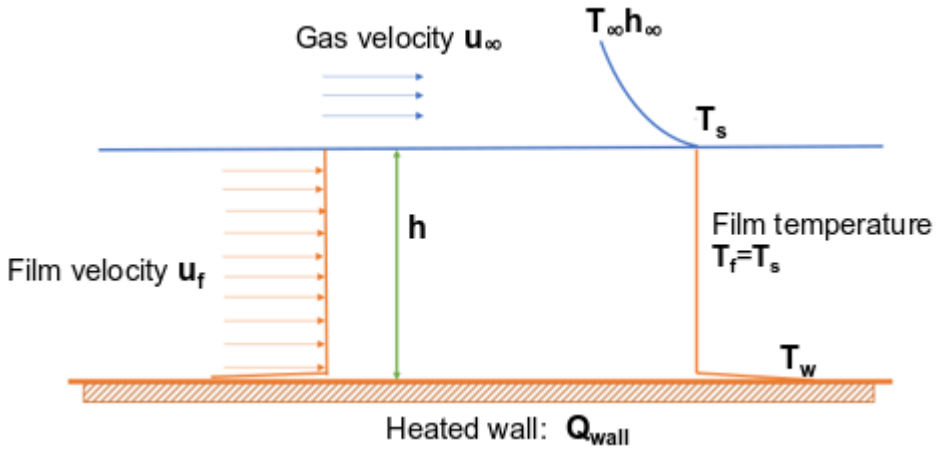
Turbulent film approximation for heat transfer

For high film Reynolds numbers, the film-to-wall heat transfer is significantly enhanced because of turbulence phenomena. Ansys Fluent provides a simplified approach that improves the heat transfer predictions when the film is in the turbulent regime.

For the heat transfer calculations only, the following assumptions are made:

- Both the transitional flow regime and the fully turbulent flow regime are defined based on the film local Reynolds number.
- For the fully turbulent flow regime, both the velocity and temperature profiles in the film are flat (see Figure 12.15: Fully Turbulent Film Flow Regime: Velocity and Temperature Profile Assumptions (p. 531)).

Figure 12.15: Fully Turbulent Film Flow Regime: Velocity and Temperature Profile Assumptions



With the Reynolds number Re expressed by

$$Re=\frac{\rho_l u_f len}{\mu_l} \tag{12.323}$$

where u_f is the film average velocity, and the remaining nomenclature is the same as in Equation 12.315 (p. 529), the Nusselt number Nu for the wall-to-film heat transfer is defined as follows:

- Fully turbulent flow regime ($Re>Re_{higher_limit}$):

$$Nu=\frac{RePr_l C_f}{1+12.7\left(Pr_l^{2/3}-1\right)\sqrt{C_f}} \tag{12.324}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$C_f = \frac{0.455/2}{\ln(0.06Re)^2} \tag{12.325}$$

The flat temperature profile is achieved by setting $f(\mathrm{Pr}) = \frac{3}{8}Pr^{-1/3} = 0$ in Equation 12.310 (p. 528) or Equation 12.311 (p. 529).

- Transitional flow regime ($Re_{lower_limit} < Re < Re_{higher_limit}$):

  The Nusselt number is defined as the largest of the Nu numbers computed by Equation 12.326 (p. 532) and either Equation 12.313 (p. 529) or Equation 12.321 (p. 530). The cubic temperature profile assumption (Equation 12.305 (p. 528)) is still maintained.

$$Nu_{tr} = Nu_{lam}\left(\frac{Re}{Re_{lower_limit}}\right)^c \tag{12.326}$$

  Here:

$$c = 0.992\ log_{10}\left(Re_{lower_limit}\right) - 3.013 \tag{12.327}$$

  and

$$Nu_{lam} = \begin{cases} 0.332Re_{lower_limit}^{1/2}Pr_l^{1/3} & \text{for constant wall temperature} \\ 0.4587Re_{lower_limit}^{1/2}Pr_l^{1/3} & \text{for constant heat flux} \end{cases} \tag{12.328}$$

  For the fully turbulent flow regime, the Nu number is set as the minimum of the values predicted by Equation 12.326 (p. 532) and Equation 12.324 (p. 531).

  The Re number limits have been calibrated with the data reported by Åkesjö et al. [12] (p. 1053) and [86] (p. 1057):

$$Re_{lower_limit} = 200{,}000$$
$$Re_{higher_limit} = 500{,}000$$

Calculation of the heat transfer coefficient from wall to film on the impingement point

The boundary layer theory is not applicable at the particle impingement point. Here, the following expression for the Nusselt number Nu is used [370] (p. 1074):

$$Nu = 0.745\,Re^{1/2}Pr_l^{1/3} \tag{12.329}$$

Here, the Reynolds number is expressed as:

$$Re = \frac{\rho_l u_p len}{\mu_l} \tag{12.330}$$

where u_p is the jet velocity (m/s).

The characteristic length len used in the above expression is the jet diameter, which is approximated by the reference diameter for the relevant injection type. The reference diameter is defined

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

as the hydraulic diameter corresponding to the surface area of the surface injection, or the maximum particle diameter for injections involving particle size distributions.

12.10. Wall Erosion

Erosion is the process by which material is removed from a wall surface due to micromechanical deformation or cracking of the wall's surface. In fluid-carrying equipment (such as gas and water turbines, pumps, heat exchangers, and so on), surface erosion is caused in part by the impact on equipment walls by solid particles entrained within a fluid flow. Eventually, wall erosion leads to equipment degradation, decreasing the performance, and reduced service life.

A wall material's resistance to erosion depends on its ability to deform under impact, among other factors listed below. With brittle materials, erosion occurs as a result of cracking and chipping away of microsized pieces of wall material, whereas with ductile materials, erosion occurs through a series of repeated micro-plastic deformations. For such materials, the highest rate of erosion usually occurs at impact angles ranging from 20° to 30° [26] (p. 1054).

The other parameters that govern the erosion mechanism are:

- Particle impact velocity
- Particle impact angle
- Mechanical properties of materials (for example, manufacturing technology, micro structural characteristics)
- Wall temperature
- Friction coefficients between particles and their target
- Shielding effects caused by rebounding particles

Experimental erosion data found in the literature are often subject to large uncertainties. Erosion rates reported by different authors can vary greatly (even by some orders of magnitude for the same material [26] (p. 1054), [27] (p. 1054)).

Note that erosion rates computed by the erosion models in Ansys Fluent are only a qualitative guide to erosion, unless precise values for the model constants are known.

The following models for predicting the erosion evolution and for determining the volume of eroded material have been adopted in Ansys Fluent:

12.10.1. Finnie Erosion Model
12.10.2. Oka Erosion Model
12.10.3. McLaury Erosion Model
12.10.4. DNV Erosion Model
12.10.5. Modeling Erosion Rates in Dense Flows
12.10.6. Accretion

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

12.10.1. Finnie Erosion Model

For nearly all ductile materials, erosion varies with impact angle and velocity according to:

$$E=k\,V_P^n\,f(\gamma) \tag{12.331}$$

where E is a dimensionless erosion mass, k is a model constant, V_P is the particle impact velocity, and $f(\gamma)$ is a dimensionless function of the impact angle γ. For metals, the value of the exponent n is generally in the range 2.3 to 2.5 radians.

Finnie's model of erosive wear relates the rate of wear to the rate of kinetic energy of impact of particles on the surface using the following functions:

$$f(\gamma)=\begin{cases}\frac{1}{3}\cos^2\gamma & \text{for } \gamma>18.5^\circ \\ \sin(2\gamma)-3\sin^2\gamma & \text{for } \gamma\le 18.5^\circ\end{cases} \tag{12.332}$$

The default constants for the Ansys Fluent Finnie Erosion model are only valid for erosion calculations where sand particles hit carbon steel walls. The value of the model constant k has been tuned to match the experiment described in [483] (p. 1081) in which 326-micron sand particles impact a carbon steel wall at a speed of 104 m/s and therefore is strictly valid only for those specific conditions.

12.10.2. Oka Erosion Model

In the Ansys Fluent implementation of the erosion model by Oka, the erosion rate E is determined as:

$$E = E_{90}\left(\frac{V}{V_{ref}}\right)^{k_2}\left(\frac{d}{d_{ref}}\right)^{k_3} f(\gamma) \tag{12.333}$$

where

E_{90} = reference erosion rate at 90° impact angle (that is, the wall material mass removed per mass of particles). Note that this quantity is usually derived by measurements (see [483] (p. 1081)).

V = particle impact velocity

V_{ref} = reference velocity

d and d_{ref} = particle diameter and particle reference diameter, respectively

k_2 and k_3 = velocity and diameter exponents, respectively

$f(\gamma)$ = impact angle function

The impact angle dependence of the erosion rate is given by:

$$f(\gamma) = (\sin\gamma)^{n_1}(1+H_v(1-\sin\gamma))^{n_2} \tag{12.334}$$

where

γ = wall impact angle (rad)

H_v = wall material Vickers hardness (GPa)

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

n_1 and n_2 = angle function constants

Table 12.3: Example of the Oka Erosion Model Constants

Constant	Sand - Steel
E_{90}	6.154e^{-4}
H_v	1.8 [GPa]
n_1	0.8
n_2	1.3
k_2	2.35
k_3	0.19
d_{ref}	326 [µm]
V_{ref}	104 [m/s]

12.10.3. McLaury Erosion Model

McLaury proposed a model for predicting the erosion rate of sand particles in water [420] (p. 1077). The model has primarily been used to simulate erosion rates in slurry erosion. The McLaury erosion rate E is determined by:

$$E = AV^n f(\gamma)$$
$$A = F\ Bh^{\mathrm{k}} \tag{12.335}$$

where

F = empirical constant

V = particle impact velocity

Bh = Brinell's hardness number of wall material

k = -0.59 for carbon steel. For other materials the exponent k is different.

with

$$f(\gamma) = b\gamma^2 + c\gamma \quad \text{for } \gamma \le \gamma_{lim} \tag{12.336}$$

$$f(\gamma) = x\ \cos^2\gamma\ \sin(w\gamma) + y\ \sin^2(\gamma) + z \quad \text{for } \gamma > \gamma_{lim} \tag{12.337}$$

The model constants b, c, w, x, and y must be determined by an experiment. z must be chosen in such a way that the angle functions (Equation 12.336 (p. 535) and Equation 12.337 (p. 535)) coincide at $\gamma = \gamma_{lim}$, where γ_{lim} is the transition angle.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Notes on the McLaury Model

The model constants given above are only valid for a suspension of water and sand. Particle impact velocities are assumed to be in the range of 0 - 10 m/s.

Table 12.4: Example of the McLaury Erosion Model Constants

Constant	Sand - Steel[a]
A	1.997e-7
γ_{lim}	15 [deg]
n	1.73
b	-13.3
c	7.85
x	1.09
y	0.125
z	automatically computed
w	1.0

[a] [420] (p. 1077)

12.10.4. DNV Erosion Model

In the DNV erosion model [4] (p. 1053), the erosion rate of particles is determined from:

$$E=k\ V_p^n f(\gamma) \tag{12.338}$$

where

n = material dependent velocity exponent

k = empirical constant [$(m/s)^{-n}$]

V_p = particle impact velocity [m/s]

$f(\gamma)$ = dimensionless function of the particle impact angle γ (in radians)

The material constants k and n are usually determined by experiments. Recommended values for two typical wall materials are listed in Table 12.5: Recommended Values for Material Constants (p. 536).

Table 12.5: Recommended Values for Material Constants

Material	k	n	Density (kg/m^3)
Steel grades	2×10^{-9}	2.6	7800
Titanium grades	2×10^{-9}	2.6	4500

The function $f(\gamma)$ for steel grades is defined by:

$$f(\gamma)=\sum_{i=1}^{8}(-1)^{i+1}A_i\gamma^i \tag{12.339}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Values for the model constants A_i in Equation 12.339 (p. 536) are given in Table 12.6: Model Constants (p. 537).

Table 12.6: Model Constants

A_1	A_2	A_3	A_4	A_5	A_6	A_7	A_8
9.370	42.295	110.864	175.804	170.137	98.398	31.211	4.170

12.10.5. Modeling Erosion Rates in Dense Flows

In dense flows, particle-particle interactions may impact the erosion rates due to the shielding effects of a dense near-wall layer of solid particles. In addition, solid particles that move nearly parallel to the wall may lead to an abrasive erosion. In Ansys Fluent, the total erosion rate in dense systems is computed as follows:

$$E_{total}=E_{abrasive}+ E_{imp} \tag{12.340}$$

where $E_{abrasive}$ is the abrasive erosion rate calculated as described in Abrasive Erosion Caused by Solid Particles (p. 537), and E_{imp} is the impact erosion rate calculated as described in Wall Shielding Effect in Dense Flow Regimes (p. 538).

The abrasive erosion and shielding are computed under the following conditions:

- a DPM phase with the **Erosion/Accretion** option enabled exists
- at least one dispersed granular Euler phase exists

In many cases with wall erosion, particles tend to concentrate in streaks in the near-wall regions. This very strongly violates the basic (and essential) DPM assumption of dispersed flow regime. To simulate densely packed particles using the Lagrange approach, particle-particle interactions as well as the volume displacement of the particles must be considered. This can be done using the DEM model in the densely packed regions, but at immense computational costs. To avoid this expense, ANSYS Fluent uses a dispersed Euler phase, which accounts for the volume displacement and any interphase interactions. The dispersed Euler phase is assumed to adequately represent the particle behavior close to the wall. The erosion is to a large extent calculated from Euler phase quantities (such as wall shear erosion). The classical impact-based erosion is reduced by blending via the shielding function.

12.10.5.1. Abrasive Erosion Caused by Solid Particles

The abrasive erosion of the solid phase in the near-wall region is modelled as a function of the solid phase wall shear stress and the solid phase near-wall velocity:

$$E_{abrasive}=AV_s^n\tau_{w,s}\frac{\alpha_s}{\alpha_{pack}} \tag{12.341}$$

where

A = empirical constant

V_s = solid phase velocity

n = velocity exponent

$\tau_{w,s}$ = wall shear stress of the solid phase

α_s = solid phase volume fraction

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

α_{pack} = packing limit (~0.63 for spherical particles)

For multiphase flows with N granular phases, the erosion rate due to wall shear stress is the accumulated erosion rate of all granular phases:

$$E_{abrasive}=\sum_{i=1}^{N}E_{abrasive,i}=\sum_{i=1}^{N}AV_{s,i}^{n}\tau_{w,s,i}\frac{\alpha_{s,i}}{\alpha_{pack}} \tag{12.342}$$

12.10.5.2. Wall Shielding Effect in Dense Flow Regimes

In the dense flow regime, particles approaching the wall may be slowed down or reflected by other solid particles that move nearly parallel to the wall in its vicinity. This shielding effect will lower the erosion rates caused by the particles impacting the solid wall surface.

To account for this effect, the following shielding function is used in Ansys Fluent:

$$f_{shield}=1-\frac{\alpha_s}{\alpha_{pack}} \tag{12.343}$$

where

α_s = volume fraction of the solid phase

α_{pack} = packing limit (~0.63 for spherical particles)

When calculating the impact erosion rate E_{imp}, the shielding function f_{shield} is used to scale the impact erosion rate for the single phase $E_{imp,sp}$:

$$E_{imp}=f_{shield}E_{imp,sp} \tag{12.344}$$

If the local solid phase volume fraction α_s approaches the packing limit α_{pack}, then the wall surface will be shielded from any impacting particles, that is, the impact erosion rate E_{imp} will be zero.

For flows with N granular phases, the shielding function is computed from the sum of the individual Eulerian volume fractions:

$$f_{shield}=1-\frac{\sum_{i=1}^{N}\alpha_{s,i}}{\alpha_{pack}} \tag{12.345}$$

12.10.6. Accretion

Particle erosion and accretion rates can be monitored at wall boundaries. The erosion rate is defined as

$$R_{erosion}=\sum_{p=1}^{N_{particles}}\frac{\dot{m}_p C(d_p) f(\gamma) v^{b(v)}}{A_{face}} \tag{12.346}$$

where $C(d_p)$ is a function of particle diameter, γ is the impact angle of the particle path with the wall face, $f(\gamma)$ is a function of impact angle, v is the relative particle velocity, $b(v)$ is a function of relative particle velocity, and A_{face} is the area of the cell face at the wall. Default values are $C=1.8\times10^{-9}$, $f=1$, and $b=0$. Note that C, f, and b are defined as boundary conditions at a wall, rather than

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

properties of a material, so the default values are not updated to reflect the actual materials being used. Therefore, you should specify values appropriate to your problem at all walls as described in Setting Particle Erosion and Accretion Parameters in the *Fluent User's Guide*. Values of these functions for sand eroding both carbon steel and aluminum are given by Edwards et al. [160] (p. 1062).

Functions for C, f, and b must be expressed as piecewise-linear, piecewise-polynomial or polynomial functions in order to define them as part of the wall boundary conditions. Therefore, it may be necessary to approximate functions found in the literature by one of these methods.

The erosion rate as calculated above is displayed in units of removed material/(area-time), that is, mass flux, and can therefore be changed accordingly to the defined units in Ansys Fluent. The functions C and f have to be specified in consistent units to build a dimensionless group with the relative particle velocity and its exponent. To compute an erosion rate in terms of length/time (mm/year, for example) you can either define a custom field function to divide the erosion rate by the density of the wall material or include this division in the units for C and/or f. Note that the units given by Ansys Fluent when displaying the erosion rate are no longer valid in the latter case.

A variety of erosion models [180] (p. 1063), [421] (p. 1077), [159] (p. 1062), [476] (p. 1080), [235] (p. 1066), [565] (p. 1085) containing model constants [235] (p. 1066), [159] (p. 1062) and angle functions can be easily implemented into Ansys Fluent. The equations describing some of the erosion models can be modified to appear in the form of the general equation describing the erosion rate, Equation 12.346 (p. 538).

For example, the Tulsa Angle Dependent Model [159] (p. 1062) described by Equation 12.347 (p. 539)

$$ER = 1.559e^{-6}B^{-0.59}F_s v^{1.73} f(\gamma) \tag{12.347}$$

can be rewritten in the form of Equation 12.346 (p. 538) with the following substitutions:

$$v^{1.73} = v^{b(v)}$$

$$1.559e^{-6}B^{-0.59}F_s = C(d_p)$$

where ER is the erosion ratio (mass of surface eroded per mass of particles impinging), B is the Brinell hardness, and F_s is a particle shape coefficient. The Tulsa model [159] (p. 1062) suggests values for F_s of 1.0 for sharp/angular sand, 0.53 for semi-rounded sand, and 0.2 for fully rounded sand. As noted above, the impact angle function can be approximated by a piecewise-polynomial fit. The following is an approximation of the impact angle function for steel being eroded by sand.

$$f(\gamma) = \begin{cases} 0 + 22.7\gamma - 38.4\gamma^2 & \gamma \le 0.267\text{rad} \\ 2.00 + 6.80\gamma - 7.50\gamma^2 + 2.25\gamma^3 & \gamma > 0.267\text{rad} \end{cases}$$

User-defined functions can be used to describe erosion models of any form. For more complex models, such as those in which f cannot be approximated by linear or polynomial functions of γ only, the default **Erosion Model** in the **Wall** boundary condition dialog box cannot be used. Hence, a user-defined function should be used instead. For information on how to apply user-defined functions for DPM erosion models, see `DEFINE_DPM_EROSION` in the Fluent Customization Manual, or contact your support engineer for further assistance.

The accretion rate is defined as

$$R_{accretion} = \sum_{p=1}^{N_{particles}} \frac{\dot{m}_p}{A_{face}} \tag{12.348}$$

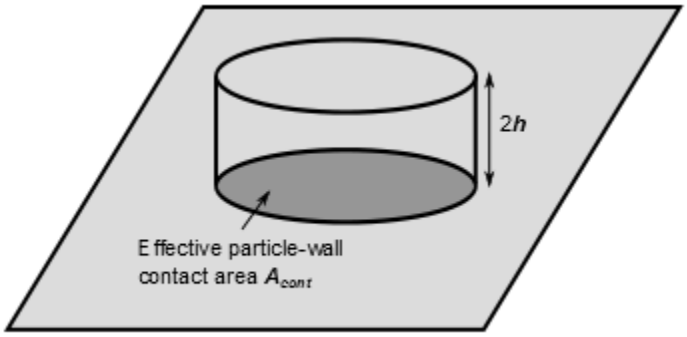
Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

12.11. Particle–Wall Impingement Heat Transfer Theory

When a liquid particle impinges a wall, the particle may deform and remain in direct contact with the wall for a short period of time before rebounding. During this time, heat is exchanged between the particle and the wall.

For droplet-to-wall heat transfer calculations, the Ansys Fluent discrete phase model assumes that the droplet deforms to a cylinder when impacting the wall (see Figure 12.16: Geometric Parameters of Deformed Impinging Droplet in Heat Transfer Calculations (p. 540)).

Figure 12.16: Geometric Parameters of Deformed Impinging Droplet in Heat Transfer Calculations



Then the heat transfer from the wall to the droplet is given as:

$$\frac{d}{dt}\left(m_p c_p T_p\right) = \frac{k_l A_{cont}}{h}\left(T_w - T_p\right) \tag{12.349}$$

where

m_p = droplet particle mass (kg)

c_p = droplet specific heat (J/kg/K)

T_p = particle temperature (K)

A_{cont} = effective particle-wall contact area (m^2)

h = particle center-point to wall distance (m)

k_l = droplet thermal conductivity (W/m/K)

T_w = wall temperature (K)

The effective particle-wall contact area A_{cont} is calculated by time-averaging the particle-wall contact area assuming a sinusoidal particle diameter variation from 0 to the maximum spreading diameter D_{max} during contact period [65] (p. 1056). The maximum spreading diameter D_{max} is computed according to [11] (p. 1053) as:

$$A_{cont} = \pi D_{max}^2 / 8 \tag{12.350}$$

$$D_{max} = 0.61 D_0 We^{0.38} \tag{12.351}$$

where

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

D_0 = droplet diameter before impact (m)

We = Impact Weber number

The contact time t_d is calculated as [65] (p. 1056):

$$t_d = f \frac{\pi}{4} \sqrt{\frac{\rho D_0^3}{\sigma}} \tag{12.352}$$

where

f = constant with the default value of 0.4

ρ = particle density (kg/m^3)

σ = surface tension (N/m)

The heat exchange between the wall and the particle is calculated by integrating Equation 12.349 (p. 540) over the contact time t_d. The increase of the droplet temperature is limited by the boiling point.

For each wall face, the particle-to-wall energy transfer Q_{imp} is computed by summing the energy contributions from all particle parcels hitting the wall face and is then added to the heat flux of the wall.

12.12. Atomizer Model Theory

In addition to the simple injection types described in Injection Types in the User's Guide, Ansys Fluent also provides more complex injection types for sprays describing primary breakup phenomena. For most types of injections, you will need to provide the initial diameter, position, and velocity of the particles. For sprays, however, there are models available to predict the droplet size and velocity distributions.

All of the atomization models use physical atomizer parameters, such as orifice diameter and mass flow rate, to calculate initial droplet size, velocity, and position.

For realistic atomizer simulations, the droplets must be randomly distributed, both spatially through a dispersion angle and in their time of release. For other types of injections in Ansys Fluent (nonatomizer), all of the droplets are released along fixed trajectories at the beginning of the time step. The atomizer models use stochastic trajectory selection and staggering to attain a random distribution. Further information on staggering can be found in section Staggering of Particles in Space and Time in the User's Guide.

Stochastic trajectory selection is the random dispersion of initial droplet directions. All of the atomizer models provide an initial dispersion angle, and the stochastic trajectory selection picks an initial direction within this angle. This approach improves the accuracy of the results for spray-dominated flows. The droplets will be more evenly spread among the computational cells near the atomizer, which improves the coupling to the gas phase by spreading drag more smoothly over the cells near the injection. Source terms in the energy and species conservation equations are also more evenly distributed among neighboring cells, improving solution convergence.

Five atomizer models are available in Ansys Fluent to predict the spray characteristics from knowledge of global parameters such as nozzle type and liquid flow rate. You can choose them as injection types

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

and define the associated parameters in the **Set Injection Properties** dialog box, as described in Injection Types in the User's Guide. Details about the atomizer models are provided below.

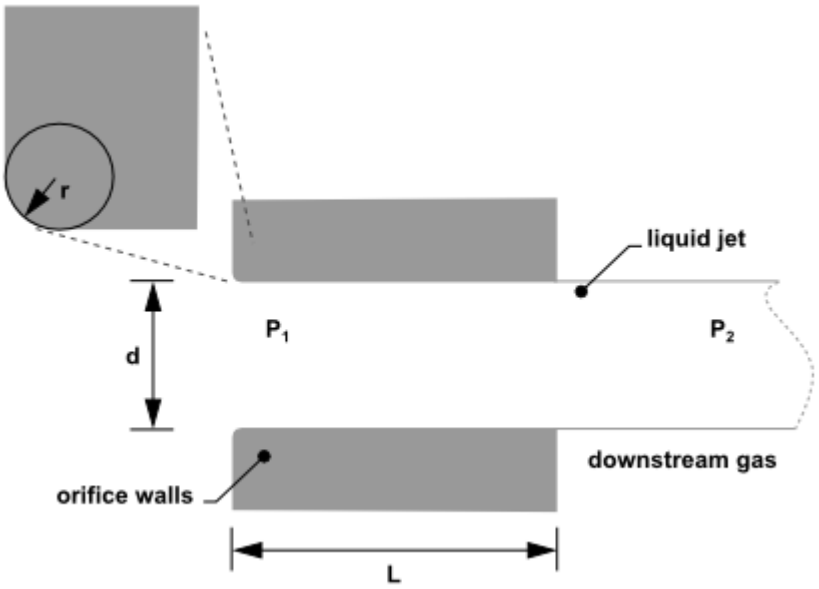
Information is organized into the following subsections:

12.12.1. The Plain-Orifice Atomizer Model
12.12.2. The Pressure-Swirl Atomizer Model
12.12.3. The Air-Blast/Air-Assist Atomizer Model
12.12.4. The Flat-Fan Atomizer Model
12.12.5. The Effervescent Atomizer Model

12.12.1. The Plain-Orifice Atomizer Model

The plain-orifice is the most common type of atomizer and the most simply made. However there is nothing simple about the physics of the internal nozzle flow and the external atomization. In the plain-orifice atomizer model in Ansys Fluent, the liquid is accelerated through a nozzle, forms a liquid jet and then breaks up to form droplets. This apparently simple process is dauntingly complex. The plain orifice may operate in three different regimes: single-phase, cavitating and flipped [615] (p. 1088). The transition between regimes is abrupt, producing dramatically different sprays. The internal regime determines the velocity at the orifice exit, as well as the initial droplet size and the angle of droplet dispersion. Diagrams of each case are shown in Figure 12.17: Single-Phase Nozzle Flow (Liquid Completely Fills the Orifice) (p. 542), Figure 12.18: Cavitating Nozzle Flow (Vapor Pockets Form Just After the Inlet Corners) (p. 543), and Figure 12.19: Flipped Nozzle Flow (Downstream Gas Surrounds the Liquid Jet Inside the Nozzle) (p. 543).

Figure 12.17: Single-Phase Nozzle Flow (Liquid Completely Fills the Orifice)



Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Figure 12.18: Cavitating Nozzle Flow (Vapor Pockets Form Just After the Inlet Corners)

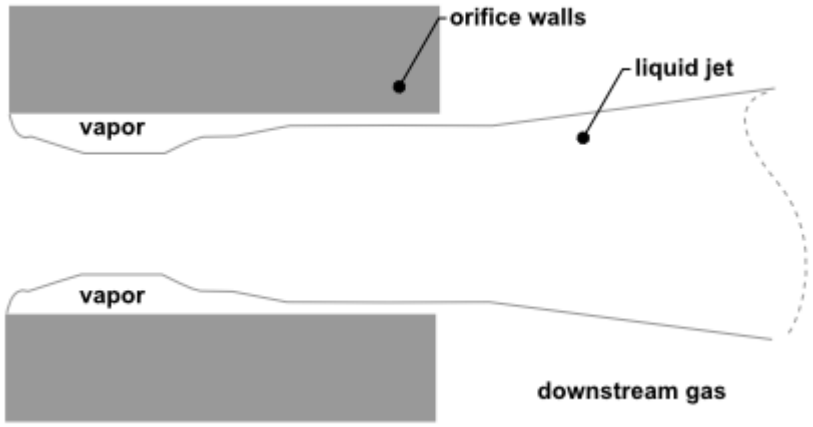
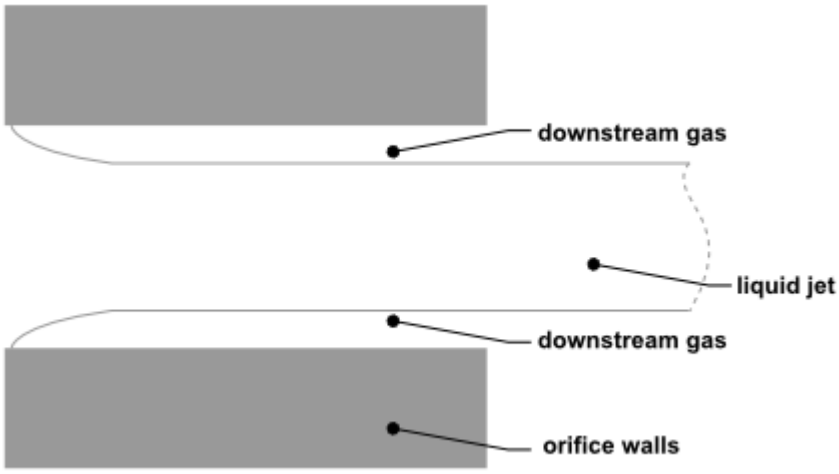


Figure 12.19: Flipped Nozzle Flow (Downstream Gas Surrounds the Liquid Jet Inside the Nozzle)



12.12.1.1. Internal Nozzle State

To accurately predict the spray characteristics, the plain-orifice model in Ansys Fluent must identify the correct state of the internal nozzle flow because the nozzle state has a tremendous effect on the external spray. Unfortunately, there is no established theory for determining the nozzle state. One must rely on empirical models obtained from experimental data. Ansys Fluent uses several dimensionless parameters to determine the internal flow regime for the plain-orifice atomizer model. These parameters and the decision-making process are summarized below.

A list of the parameters that control internal nozzle flow is given in Table 12.7: List of Governing Parameters for Internal Nozzle Flow (p. 543). These parameters may be combined to form nondimensional characteristic lengths such as *r/d* and *L/d*, as well as nondimensional groups like the Reynolds number based on hydraulic "head" Re_h and the cavitation parameter K [479] (p. 1081).

Table 12.7: List of Governing Parameters for Internal Nozzle Flow

nozzle diameter	d
nozzle length	L

radius of curvature of the inlet corner	r
upstream pressure	p_1
downstream pressure	p_2
viscosity	μ
liquid density	ρ_l
vapor pressure	p_v

$$Re_h = \frac{d\rho_l}{\mu}\sqrt{\frac{2\left(p_1 - p_2\right)}{\rho_l}} \tag{12.353}$$

$$K = \frac{p_1 - p_v}{p_1 - p_2} \tag{12.354}$$

The liquid flow often contracts in the nozzle, as can be seen in Figure 12.18: Cavitating Nozzle Flow (Vapor Pockets Form Just After the Inlet Corners) (p. 543) and Figure 12.19: Flipped Nozzle Flow (Downstream Gas Surrounds the Liquid Jet Inside the Nozzle) (p. 543). Nurick [479] (p. 1081) found it helpful to use a coefficient of contraction (C_c) to represent the reduction in the cross-sectional area of the liquid jet. The coefficient of contraction is defined as the area of the stream of contracting liquid over the total cross-sectional area of the nozzle. Ansys Fluent uses Nurick's fit for the coefficient of contraction:

$$C_c = \frac{1}{\sqrt{\frac{1}{C_{ct}^2} - \frac{11.4r}{d}}} \tag{12.355}$$

Here, C_{ct} is a theoretical constant equal to 0.611, which comes from potential flow analysis of flipped nozzles.

12.12.1.2. Coefficient of Discharge

Another important parameter for describing the performance of nozzles is the coefficient of discharge (C_d). The coefficient of discharge is the ratio of the mass flow rate through the nozzle to the theoretical maximum mass flow rate:

$$C_d = \frac{\dot{m}_{eff}}{A\sqrt{2\rho_l\left(p_1 - p_2\right)}} \tag{12.356}$$

where $\dot{m}_{eff}$ is the *effective* mass flow rate of the nozzle, defined by

$$\dot{m}_{eff} = \frac{2\pi\dot{m}}{\Delta\varphi} \tag{12.357}$$

Here, $\dot{m}$ is the mass flow rate specified in the user interface, and $\Delta\varphi$ is the difference between the azimuthal stop angle and the azimuthal start angle

$$\Delta\varphi = \varphi_{stop} - \varphi_{start} \tag{12.358}$$

as input by you (see Point Properties for Plain-Orifice Atomizer Injections in the User's Guide). Note that the mass flow rate that you input should be for the appropriate start and stop angles, in other

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

words the correct mass flow rate for the sector being modeled. Note also that for $\Delta\varphi$ of 2π, the effective mass flow rate is identical to the mass flow rate in the interface.

The cavitation number (K in Equation 12.354 (p. 544)) is an essential parameter for predicting the inception of cavitation. The inception of cavitation is assumed to occur at a value given by the following empirical relationship:

$$K_{incep}=2\left(1-\frac{r}{d}\right)^{2}-\frac{1000}{Re_h} \tag{12.359}$$

Similarly, a critical value of K where flip occurs is given by

$$K_{crit}=1+\frac{1}{\left(1+\frac{L}{4d}\right)\left(1+\frac{2000}{Re_h}\right)e^{70r/d}} \tag{12.360}$$

If r/d is greater than 0.05, then flip is deemed impossible and K_{crit} is set to 1.0.

The cavitation number, K, is compared to the values of K_{incep} and K_{crit} to identify the nozzle state. The decision tree is shown in Figure 12.20: Decision Tree for the State of the Cavitating Nozzle (p. 545). Depending on the state of the nozzle, a unique closure is chosen for the above equations.

For a single-phase nozzle ($K>K_{incep}, K\geq K_{crit}$) [367] (p. 1074), the coefficient of discharge is given by

$$C_d=\frac{1}{\frac{1}{C_{du}}+20\frac{(1+2.25L/d)}{Re_h}} \tag{12.361}$$

where C_{du} is the ultimate discharge coefficient, and is defined as

$$C_{du}=0.827-0.0085\frac{L}{d} \tag{12.362}$$

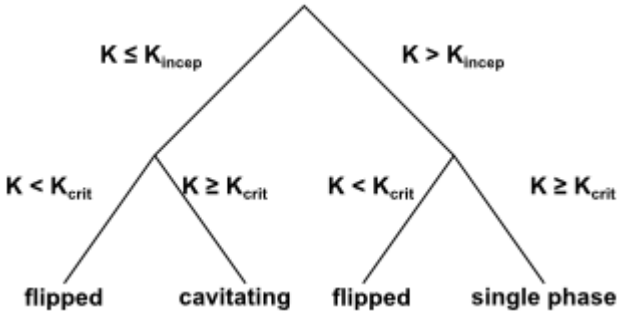
For a cavitating nozzle ($K_{crit}\leq K\leq K_{incep}$) [479] (p. 1081) the coefficient of discharge is determined from

$$C_d=C_c\sqrt{K} \tag{12.363}$$

For a flipped nozzle ($K<K_{crit}$) [479] (p. 1081),

$$C_d=C_{ct}=0.611 \tag{12.364}$$

Figure 12.20: Decision Tree for the State of the Cavitating Nozzle



All of the nozzle flow equations are solved iteratively, along with the appropriate relationship for coefficient of discharge as given by the nozzle state. The nozzle state may change as the upstream

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

or downstream pressures change. Once the nozzle state is determined, the exit velocity is calculated, and appropriate correlations for spray angle and initial droplet size distribution are determined.

12.12.1.3. Exit Velocity

For a single-phase nozzle, the estimate of exit velocity (u) comes from the conservation of mass and the assumption of a uniform exit velocity:

$$u=\frac{\dot{m}_{eff}}{\rho_l A} \tag{12.365}$$

For the cavitating nozzle, Schmidt and Corradini [578] (p. 1086) have shown that the uniform exit velocity is not accurate. Instead, they derived an expression for a higher velocity over a reduced area:

$$u=\frac{2C_c p_1-p_2+\left(1-2C_c\right)p_v}{C_c\sqrt{2\rho_l\left(p_1-p_v\right)}} \tag{12.366}$$

This analytical relation is used for cavitating nozzles in Ansys Fluent. For the case of flipped nozzles, the exit velocity is found from the conservation of mass and the value of the reduced flow area:

$$u=\frac{\dot{m}_{eff}}{\rho_l C_{ct} A} \tag{12.367}$$

12.12.1.4. Spray Angle

The correlation for the spray angle (θ) comes from the work of Ranz [538] (p. 1084):

$$\frac{\theta}{2}=\begin{cases}\tan^{-1}\left[\frac{4\pi}{C_A}\sqrt{\frac{\rho_g}{\rho_l}}\frac{\sqrt{3}}{6}\right] & single phase, cavitating \\ 0.01 & flipped\end{cases} \tag{12.368}$$

The spray angle for both single-phase and cavitating nozzles depends on the ratio of the gas and liquid densities and also the parameter C_A. For flipped nozzles, the spray angle has a constant value.

The parameter C_A, which you must specify, is thought to be a constant for a given nozzle geometry. The larger the value, the narrower the spray. Reitz [545] (p. 1084) suggests the following correlation for C_A:

$$C_A=3+\frac{L}{3.6d} \tag{12.369}$$

The spray angle is sensitive to the internal flow regime of the nozzle. Hence, you may want to choose smaller values of C_A for cavitating nozzles than for single-phase nozzles. Typical values range from 4.0 to 6.0. The spray angle for flipped nozzles is a small, arbitrary value that represents the lack of any turbulence or initial disturbance from the nozzle.

12.12.1.5. Droplet Diameter Distribution

One of the basic characteristics of an injection is the distribution of drop size. For an atomizer, the droplet diameter distribution is closely related to the nozzle state. Ansys Fluent's spray models use a two-parameter Rosin-Rammler distribution, characterized by the most probable droplet size and a spread parameter. The most probable droplet size, d_0 is obtained in Ansys Fluent from the Sauter

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

mean diameter, d_{32} [349] (p. 1073). For more information about the Rosin-Rammler size distribution, see Using the Rosin-Rammler Diameter Distribution Method in the User's Guide.

For single-phase nozzle flows, the correlation of Wu et al. [716] (p. 1094) is used to calculate d_{32} and relate the initial drop size to the estimated turbulence quantities of the liquid jet:

$$d_{32}=133.0\lambda We^{-0.74}, \tag{12.370}$$

where $\lambda=d/8$, λ is the radial integral length scale at the jet exit based upon fully-developed turbulent pipe flow, and We is the Weber number, defined as

$$We\equiv\frac{\rho_l u^2\lambda}{\sigma}. \tag{12.371}$$

Here, σ is the droplet surface tension. For a more detailed discussion of droplet surface tension and the Weber number, see Secondary Breakup Model Theory (p. 554). For more information about mean particle diameters, see Summary Reporting of Current Particles in the User's Guide.

For cavitating nozzles, Ansys Fluent uses a slight modification of Equation 12.370 (p. 547). The initial jet diameter used in Wu's correlation, d, is calculated from the effective area of the cavitating orifice exit, and therefore represents the effective diameter of the exiting liquid jet, d_{eff}. For an explanation of effective area of cavitating nozzles, see Schmidt and Corradini [578] (p. 1086).

The length scale for a cavitating nozzle is $\lambda=d_{eff}/8$, where

$$d_{eff}=\sqrt{\frac{4\dot{m}_{eff}}{\pi\rho_l u}}. \tag{12.372}$$

For the case of the flipped nozzle, the initial droplet diameter is set to the diameter of the liquid jet:

$$d_0=d\sqrt{C_{ct}} \tag{12.373}$$

where d_0 is defined as the most probable diameter.

The second parameter required to specify the droplet size distribution is the spread parameter, s. The values for the spread parameter are chosen from past modeling experience and from a review of experimental observations. Table 12.8: Values of Spread Parameter for Different Nozzle States (p. 547) lists the values of s for the three nozzle states. The larger the value of the spread parameter, the narrower the droplet size distribution.

Table 12.8: Values of Spread Parameter for Different Nozzle States

State	Spread Parameter
single phase	3.5
cavitating	1.5
flipped	∞

Since the correlations of Wu et al. provide the Sauter mean diameter, d_{32}, these are converted to the most probable diameter, d_0. Lefebvre [349] (p. 1073) gives the most general relationship between

the Sauter mean diameter and most probable diameter for a Rosin-Rammler distribution. The simplified version for s=3.5 is as follows:

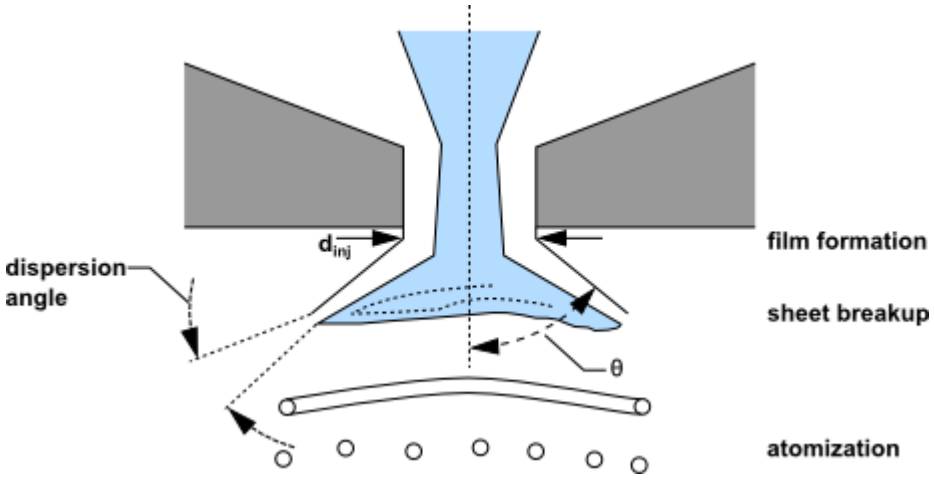
$$d_0 = 1.2726 d_{32}\left(1-\frac{1}{s}\right)^{1/s} \tag{12.374}$$

At this point, the droplet size, velocity, and spray angle have been determined and the initialization of the injections is complete.

12.12.2. The Pressure-Swirl Atomizer Model

Another important type of atomizer is the pressure-swirl atomizer, sometimes referred to by the gas-turbine community as a simplex atomizer. This type of atomizer accelerates the liquid through nozzles known as swirl ports into a central swirl chamber. The swirling liquid pushes against the walls of the swirl chamber and develops a hollow air core. It then emerges from the orifice as a thinning sheet, which is unstable, breaking up into ligaments and droplets. The pressure-swirl atomizer is very widely used for liquid-fuel combustion in gas turbines, oil furnaces, and direct-injection spark-ignited automobile engines. The transition from internal injector flow to fully-developed spray can be divided into three steps: film formation, sheet breakup, and atomization. A sketch of how this process is thought to occur is shown in Figure 12.21: Theoretical Progression from the Internal Atomizer Flow to the External Spray (p. 548).

Figure 12.21: Theoretical Progression from the Internal Atomizer Flow to the External Spray



The interaction between the air and the sheet is not well understood. It is generally accepted that an aerodynamic instability causes the sheet to break up. The mathematical analysis below assumes that Kelvin-Helmholtz waves grow on the sheet and eventually break the liquid into ligaments. It is then assumed that the ligaments break up into droplets due to varicose instability. Once the liquid droplets are formed, the spray evolution is determined by drag, collision, coalescence, and secondary breakup.

The pressure-swirl atomizer model used in Ansys Fluent is called the Linearized Instability Sheet Atomization (LISA) model of Schmidt et al. [580] (p. 1086). The LISA model is divided into two stages:

1. film formation

2. sheet breakup and atomization

Both parts of the model are described below.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

12.12.2.1. Film Formation

The centrifugal motion of the liquid within the injector creates an air core surrounded by a liquid film. The thickness of the film at the injector exit, t, is related to the mass flow rate by:

$$\dot{m}_{eff}=\pi\rho_l u t\left(d_{inj}-t\right) \tag{12.375}$$

where d_{inj} is the injector exit diameter, and $\dot{m}_{eff}$ is the effective mass flow rate, which is defined by Equation 12.357 (p. 544). The other unknown in Equation 12.375 (p. 549) is u, the axial component of velocity at the injector exit. This quantity depends on internal details of the injector and is difficult to calculate from first principles. Instead, the approach of Han et al. [229] (p. 1066) is used. The total velocity is assumed to be related to the injector pressure by

$$U=k_v\sqrt{\frac{2\Delta p}{\rho_l}} \tag{12.376}$$

where k_v is the velocity coefficient. Lefebvre [349] (p. 1073) has noted that k_v is a function of the injector design and injection pressure. If the swirl ports are treated as nozzles and if it is assumed that the dominant portion of the pressure drop occurs at those ports, k_v is the expression for the discharge coefficient (C_d). For single-phase nozzles with sharp inlet corners and $L/\ d$ ratios of 4, a typical C_d value is 0.78 or less [367] (p. 1074). If the nozzles are cavitating, the value of C_d may be as low as 0.61. Hence, 0.78 could be considered a practical upper bound for k_v. The effect of other momentum losses is approximated by reducing k_v by 10% to 0.7.

Physical limits on k_v require that it be less than unity from conservation of energy, yet be large enough to permit sufficient mass flow. The requirement that the size of the air-core be non-negative implies the following constraint on the film thickness, t:

$$\pi t\left(d_{inj}-t\right)\leq\tfrac{\pi}{4}d_{inj}^2 \tag{12.377}$$

Combining this with Equation 12.375 (p. 549) gives the following constraint on the axial velocity, u:

$$u\geq\frac{4\dot{m}_{eff}}{\pi d_{inj}^2\rho_l} \tag{12.378}$$

This can be combined with Equation 12.376 (p. 549) and $u=U\cos\theta$, where θ is the spray half-angle and is assumed to be known. This yields a constraint on k_v:

$$k_v\geq\frac{4\dot{m}_{eff}}{\pi d_{inj}^2\rho_l\cos\theta}\sqrt{\frac{\rho_l}{2\Delta p}} \tag{12.379}$$

Thus, Fluent uses the following expression for k_v:

$$k_v=\max\left[0.7,\frac{4\dot{m}_{eff}}{\pi d_{inj}^2\rho_l\cos\theta}\sqrt{\frac{\rho_l}{2\Delta p}}\right] \tag{12.380}$$

Assuming that Δp is known, Equation 12.376 (p. 549) can be used to find U. Once U is determined, u is found from

$$u=U\cos\theta \tag{12.381}$$

At this point, the thickness and axial component of the liquid film are known at the injector exit. The tangential component of velocity ($w=U\sin\theta$) is assumed to be equal to the radial velocity

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

component of the liquid sheet downstream of the nozzle exit. The axial component of velocity is assumed to remain constant.

12.12.2.2. Sheet Breakup and Atomization

The pressure-swirl atomizer model includes the effects of the surrounding gas, liquid viscosity, and surface tension on the breakup of the liquid sheet. Details of the theoretical development of the model are given in Senecal et al. [584] (p. 1087) and are only briefly presented here. For a more robust implementation, the gas-phase velocity is neglected in calculating the relative liquid-gas velocity and is instead set by you. This avoids having the injector parameters depend too heavily on the usually under-resolved gas-phase velocity field very near the injection location.

The model assumes that a two-dimensional, viscous, incompressible liquid sheet of thickness $2h$ moves with velocity U through a quiescent, inviscid, incompressible gas medium. The liquid and gas have densities of ρ_l and ρ_g, respectively, and the viscosity of the liquid is μ_l. A coordinate system is used that moves with the sheet, and a spectrum of infinitesimal wavy disturbances of the form

$$\eta=\eta_0 e^{ikx+\omega t} \tag{12.382}$$

is imposed on the initially steady motion. The spectrum of disturbances results in fluctuating velocities and pressures for both the liquid and the gas. In Equation 12.382 (p. 550), η_0 is the initial wave amplitude, $k=2\pi/\lambda$ is the wave number, and $\omega=\omega_r+i\omega_i$ is the complex growth rate. The most unstable disturbance has the largest value of ω_r, denoted here by Ω, and is assumed to be responsible for sheet breakup. Thus, it is desired to obtain a dispersion relation $\omega=\omega(k)$ from which the most unstable disturbance can be calculated as a function of wave number.

Squire [625] (p. 1089), Li and Tankin [363] (p. 1074), and Hagerty and Shea [224] (p. 1066) have shown that two solutions, or modes, exist that satisfy the governing equations subject to the boundary conditions at the upper and lower interfaces. The first solution, called the sinuous mode, has waves at the upper and lower interfaces in phase. The second solution is called the varicose mode, which has the waves at the upper and lower interfaces π radians out of phase. It has been shown by numerous authors (for example, Senecal et al. [584] (p. 1087)) that the sinuous mode dominates the growth of varicose waves for low velocities and low gas-to-liquid density ratios. In addition, it can be shown that the sinuous and varicose modes become indistinguishable for high-velocity flows. As a result, the atomization model in Ansys Fluent is based upon the growth of sinuous waves on the liquid sheet.

As derived in Senecal et al. [584] (p. 1087), the dispersion relation for the sinuous mode is given by

$$\omega^2\left[\tanh(kh)+Q\right]+\left[4\nu_l k^2\tanh(kh)+2iQkU\right]+ \tag{12.383}$$

$$4\nu_l k^4\tanh(kh)-4\nu_l^2 k^3\ell\tanh(\ell h)-QU^2k^2+\frac{\sigma k^3}{\rho_l}=0 \tag{12.384}$$

where $Q=\rho_g/\rho_l$ and $\ell^2=k^2+\omega/\nu_l$.

Above a critical Weber number of We_g = 27/16 (based on the liquid velocity, gas density, and sheet half-thickness), the fastest-growing waves are short. For $We_g<27/16$, the wavelengths are long compared to the sheet thickness. The speed of modern high pressure fuel injection systems is high enough to ensure that the film Weber number is well above this critical limit.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

An order-of-magnitude analysis using typical values shows that the terms of second order in viscosity can be neglected in comparison to the other terms in Equation 12.384 (p. 550). Using this assumption, Equation 12.384 (p. 550) reduces to

$$\omega_r = \frac{1}{\tanh(kh)+Q}\left\{-2\nu_l k^2 \tanh(kh) + \right. \tag{12.385}$$

$$\left. \sqrt{4\nu_l^2 k^4 \tanh^2(kh) - Q^2 U^2 k^2 - [\tanh(kh)+Q]\left[-QU^2k^2 + \frac{\sigma k^3}{\rho_l}\right]}\right\} \tag{12.386}$$

For waves that are long compared with the sheet thickness, a mechanism of sheet disintegration proposed by Dombrowski and Johns [145] (p. 1061) is adopted. For long waves, ligaments are assumed to form from the sheet breakup process once the unstable waves reach a critical amplitude. If the surface disturbance has reached a value of η_b at breakup, a breakup time, τ, can be evaluated:

$$\eta_b = \eta_0 e^{\Omega\tau} \Rightarrow \frac{1}{\Omega}\ln\left(\frac{\eta_b}{\eta_0}\right) \tag{12.387}$$

where Ω, the maximum growth rate, is found by numerically maximizing Equation 12.386 (p. 551) as a function of k. The maximum is found using a binary search that checks the sign of the derivative. The sheet breaks up and ligaments will be formed at a length given by

$$L_b = U\tau = \frac{U}{\Omega}\ln\left(\frac{\eta_b}{\eta_0}\right) \tag{12.388}$$

where the quantity $\ln\left(\eta_b/\eta_o\right)$ is an empirical *sheet* constant that you must specify. The default value of 12 was obtained theoretically by Weber [692] (p. 1093) for liquid jets. Dombrowski and Hooper [144] (p. 1061) showed that a value of 12 for the sheet constant agreed favorably with experimental sheet breakup lengths over a range of Weber numbers from 2 to 200.

The diameter of the ligaments formed at the point of breakup can be obtained from a mass balance. If it is assumed that the ligaments are formed from tears in the sheet twice per wavelength, the resulting diameter is given by

$$d_L = \sqrt{\frac{8h}{K_s}} \tag{12.389}$$

where K_s is the wave number corresponding to the maximum growth rate, Ω. The ligament diameter depends on the sheet thickness, which is a function of the breakup length. The film thickness is calculated from the breakup length and the radial distance from the center line to the mid-line of the sheet at the atomizer exit, r_0:

$$h_{end} = \frac{r_0 h_0}{r_0 + L_b \sin\left(\frac{\theta}{2}\right)} \tag{12.390}$$

This mechanism is not used for waves that are short compared to the sheet thickness. For short waves, the ligament diameter is assumed to be linearly proportional to the wavelength that breaks up the sheet,

$$d_L = \frac{2\pi C_L}{K_s} \tag{12.391}$$

where C_L, or the ligament constant, is equal to 0.5 by default.

In either the long-wave or the short-wave case, the breakup from ligaments to droplets is assumed to behave according to Weber's [692] (p. 1093) analysis for capillary instability.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$d_0 = 1.88 d_L \left(1 + 3Oh\right)^{1/6} \tag{12.392}$$

Here, Oh is the Ohnesorge number, which is a combination of the Reynolds number and the Weber number (see Jet Stability Analysis (p. 560) for more details about Oh). Once d_0 has been determined from Equation 12.392 (p. 552), it is assumed that this droplet diameter is the most probable droplet size of a Rosin-Rammler distribution with a spread parameter of 3.5 and a default dispersion angle of 6° (which can be modified in the user interface). The choice of spread parameter and dispersion angle is based on past modeling experience [579] (p. 1086). It is important to note that the spray cone angle must be specified by you when using this model.

12.12.3. The Air-Blast/Air-Assist Atomizer Model

In order to accelerate the breakup of liquid sheets from an atomizer, an additional air stream is often directed through the atomizer. The liquid is formed into a sheet by a nozzle, and air is then directed against the sheet to promote atomization. This technique is called air-assisted atomization or air-blast atomization, depending on the quantity of air and its velocity. The addition of the external air stream past the sheet produces smaller droplets than without the air. Though the exact mechanism for this enhanced performance is not completely understood, it is thought that the assisting air may accelerate the sheet instability. The air may also help disperse the droplets, preventing collisions between them. Air-assisted atomization is used in many of the same applications as pressure-swirl atomization, where especially fine atomization is required.

Ansys Fluent's air-blast atomization model is a variation of the pressure-swirl model. One important difference between them is that the sheet thickness is set directly in the air-blast atomizer model. This input is necessary because of the variety of sheet formation mechanisms used in air-blast atomizers. Hence the air-blast atomizer model does not contain the sheet formation equations that were included in the pressure-swirl atomizer model (Equation 12.375 (p. 549) – Equation 12.381 (p. 549)). You will also specify the maximum relative velocity that is produced by the sheet and air. Though this quantity could be calculated, specifying a value relieves you from the necessity of finely resolving the atomizer internal flow. This feature is convenient for simulations in large domains, where the atomizer is very small by comparison.

An additional difference is that the air-blast atomizer model assumes that the sheet breakup is always due to short waves. This assumption is a consequence of the greater sheet thickness commonly found in air-blast atomizers. Hence the ligament diameter is assumed to be linearly proportional to the wavelength of the fastest-growing wave on the sheet, and is calculated from Equation 12.391 (p. 551).

Other inputs are similar to the pressure-swirl model – you must provide the mass flow rate and spray angle. The angle in the case of the air-blast atomizer is the initial trajectory of the film as it leaves the end of the orifice. Specification of the inner and outer diameters of the film at the atomizer exit are also required, in addition to the dispersion angle whose default value is 6° (which can be modified in the user interface).

Since the air-blast atomizer model does not include internal gas flows, you must create the air streams surrounding the injector as boundary conditions within the Ansys Fluent simulation. These streams are ordinary continuous-phase flows and require no special treatment.

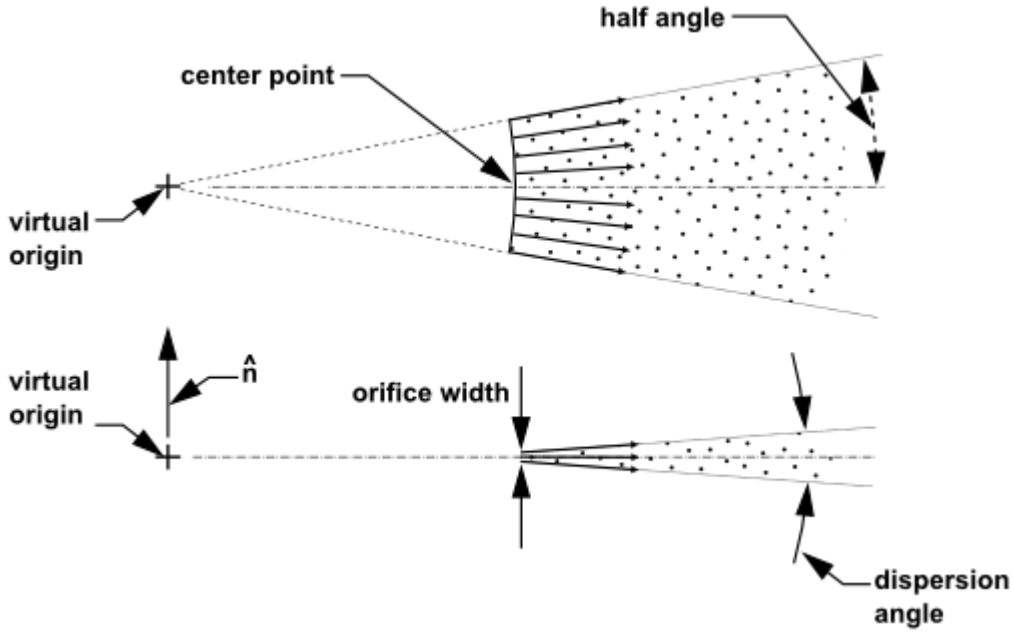
Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

12.12.4. The Flat-Fan Atomizer Model

The flat-fan atomizer is very similar to the pressure-swirl atomizer, but it makes a flat sheet and does not use swirl. The liquid emerges from a wide, thin orifice as a flat liquid sheet that breaks up into droplets. The primary atomization process is thought to be similar to the pressure-swirl atomizer. Some researchers believe that flat-fan atomization, because of jet impingement, is very similar to the atomization of a flat sheet. The flat-fan model could serve doubly for this application.

The flat-fan atomizer is available only for 3D models. An image of the three-dimensional flat fan is shown in Figure 12.22: Flat Fan Viewed from Above and from the Side (p. 553). The model assumes that the fan originates from a virtual origin. You need to provide the location of this origin, which is the intersection of the lines that mark the sides of the fan as well as the location of the center point of the arc from which the fan originates. Ansys Fluent will find the vector that points from the origin to the center point in order to determine the direction of the injection. You also need to provide the half-angle of the fan arc, the width of the orifice (in the normal direction) and the mass flow rate of the liquid to use the flat-fan atomizer model.

Figure 12.22: Flat Fan Viewed from Above and from the Side



The breakup of the flat fan is calculated very much like the breakup of the sheet in the pressure-swirl atomizer. The sheet breaks up into ligaments that then form individual droplets. The only difference is that for short waves, the flat fan sheet is assumed to form ligaments at half-wavelength intervals. Hence the ligament diameter for short waves is given by

$$d_L = \sqrt{\frac{16h}{K_s}} \tag{12.393}$$

In this case, d_L in Equation 12.393 (p. 553) is taken to be the most probable diameter, with a Rosin-Rammler spread parameter of 3.5 and a default dispersion angle of 6°. This angle can be set in the **Set Injection Properties** dialog box. In all other respects, the flat-fan atomizer model is like the sheet breakup portion of the pressure-swirl atomizer.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

12.12.5. The Effervescent Atomizer Model

Effervescent atomization is the injection of liquid infused with a super-heated (with respect to downstream conditions) liquid or propellant. As the volatile liquid exits the nozzle, it rapidly changes phase. This phase change quickly breaks up the stream into small droplets with a wide dispersion angle. The model also applies to cases where a very hot liquid is discharged.

Since the physics of effervescence is not well understood, the model must rely on rough empirical fits. The photographs of Reitz and Bracco [545] (p. 1084) provide some insights. These photographs show a dense liquid core to the spray, surrounded by a wide shroud of smaller droplets.

The initial velocity of the droplets is computed from conservation of mass, assuming the exiting jet has a cross-sectional area that is C_{ct} times the nozzle area, where C_{ct} is a fixed constant, equal to `0.611` as specified in Equation 12.355 (p. 544) and Equation 12.364 (p. 545).

$$u=\frac{\dot{m}_{eff}}{\rho_l C_{ct} A} \tag{12.394}$$

The maximum droplet diameter is set to the effective diameter of the exiting jet:

$$d_{max}=d\sqrt{C_{ct}} \tag{12.395}$$

The droplet size is then sampled from a Rosin-Rammler distribution with a spread parameter of 4.0. (See Using the Rosin-Rammler Diameter Distribution Method in the User's Guide for details on the Rosin-Rammler distribution.) The most probable droplet size depends on the angle, θ, between the droplet's stochastic trajectory and the injection direction:

$$d_0=d_{max}e^{-\left(\theta/\Theta_s\right)^2} \tag{12.396}$$

The dispersion angle multiplier, Θ_s, is computed from the quality, x, and the specified value for the dispersion constant, C_{eff}:

$$x = \frac{\dot{m}_{\text{vapor}}}{\left(\dot{m}_{\text{vapor}}+\dot{m}_{\text{liquid}}\right)} \tag{12.397}$$

$$\Theta s = \frac{x}{C_{eff}} \tag{12.398}$$

This technique creates a spray with large droplets in the central core and a shroud of smaller surrounding droplets. The droplet temperature is initialized to 0.99 times the saturation temperature, such that the temperature of the droplet is close to boiling. To complete the model, the flashing vapor must also be included in the calculation. This vapor is part of the continuous phase and not part of the discrete phase model. You must create an inlet at the point of injection when you specify boundary conditions for the continuous phase. When the effervescent atomizer model is selected, you will need to specify the nozzle diameter, mass flow rate, mixture quality, saturation temperature of the volatile substance, spray half-angle and dispersion constant in addition to specifying the position and direction of the injector.

12.13. Secondary Breakup Model Theory

Ansys Fluent offers the following droplet breakup models:

- Taylor Analogy Breakup (TAB) model

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

The TAB model is recommended for low-Weber-number injections and is well suited for low-speed sprays into a standard atmosphere.

- Wave model

  The Wave model is more applicable for Weber numbers greater than 100.

- KHRT model

  The KHRT offers a transition between the above application areas. It considers the competing effects of aerodynamic breakup and instabilities due to droplet acceleration. It is combined with the Levich model for the liquid core length.

- Stochastic Secondary Droplet (SSD) model

  The SSD model treats breakup at high Weber numbers as a discrete random event resulting in a distribution of diameter scales over a range.

- Madabhushi breakup model

  The Madabhushi breakup model is suitable for numerical simulations of liquid jets in subsonic crossflow.

- Schmehl breakup model

  The Schmehl breakup model uses an advanced modeling of secondary breakup of droplets including bag, multimode, and shear mechanisms for improved physical description of the spray.

The Wave and KHRT models are popular for use in high-speed fuel-injection applications. The choice of model depends on the relative importance of momentum and surface tension in the droplet breakup; this is described by the Weber number, which is defined in Equation 12.406 (p. 557).

Details for each model are provided in the following sections:

12.13.1. Taylor Analogy Breakup (TAB) Model
12.13.2. Wave Breakup Model
12.13.3. KHRT Breakup Model
12.13.4. Stochastic Secondary Droplet (SSD) Model
12.13.5. Madabhushi Breakup Model
12.13.6. Schmehl Breakup Model

12.13.1. Taylor Analogy Breakup (TAB) Model

12.13.1.1. Introduction

The Taylor analogy breakup (TAB) model is a classic method for calculating droplet breakup, which is applicable to many engineering sprays. This method is based upon Taylor's analogy [644] (p. 1090)

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

between an oscillating and distorting droplet and a spring mass system. Table 12.9: Comparison of a Spring-Mass System to a Distorting Droplet (p. 556) illustrates the analogous components.

Table 12.9: Comparison of a Spring-Mass System to a Distorting Droplet

Spring-Mass System	Distorting and Oscillating Droplet
restoring force of spring	surface tension forces
external force	droplet drag force
damping force	droplet viscosity forces

The resulting TAB model equation set, which governs the oscillating and distorting droplet, can be solved to determine the droplet oscillation and distortion at any given time. As described in detail below, when the droplet oscillations grow to a critical value the "parent" droplet will break up into a number of smaller "child" droplets. As a droplet is distorted from a spherical shape, the drag coefficient changes. A drag model that incorporates the distorting droplet effects is available in Ansys Fluent. See Dynamic Drag Model Theory (p. 459) for details.

12.13.1.2. Use and Limitations

The TAB model is best for low-Weber-number sprays. Extremely high-Weber-number sprays result in shattering of droplets, which is not described well by the spring-mass analogy.

12.13.1.3. Droplet Distortion

The equation governing a damped, forced oscillator is [491] (p. 1081)

$$F - kx - d\frac{dx}{dt} = m\frac{d^2x}{dt^2} \tag{12.399}$$

where x is the displacement of the droplet equator from its spherical (undisturbed) position. The coefficients of this equation are taken from Taylor's analogy:

$$\frac{F}{m} = C_F\frac{\rho_g u^2}{\rho_l r} \tag{12.400}$$

$$\frac{k}{m} = C_k\frac{\sigma}{\rho_l r^3} \tag{12.401}$$

$$\frac{d}{m} = C_d\frac{\mu_l}{\rho_l r^2} \tag{12.402}$$

where ρ_l and ρ_g are the discrete phase and continuous phase densities, u is the relative velocity of the droplet, r is the undisturbed droplet radius, σ is the droplet surface tension, and μ_l is the droplet viscosity. The dimensionless constants C_F, C_k, and C_d will be defined later.

The droplet is assumed to break up if the distortion grows to a critical ratio of the droplet radius. This breakup requirement is given as

$$x > C_b r \tag{12.403}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Where C_b is a constant equal to 0.5, if breakup is assumed to occur when the distortion is equal to half the droplet radius, that is, oscillations at the north and south pole with this amplitude will meet at the droplet center. This implicitly assumes that the droplet is undergoing only one (fundamental) oscillation mode. Equation 12.399 (p. 556) is nondimensionalized by setting $y=x/\left(C_b r\right)$ and substituting the relationships in Equation 12.400 (p. 556) – Equation 12.402 (p. 556):

$$\frac{d^2y}{dt^2}=\frac{C_F}{C_b}\frac{\rho_g}{\rho_l}\frac{u^2}{r^2}-\frac{C_k\sigma}{\rho_l r^3}y-\frac{C_d\mu_l}{\rho_l r^2}\frac{dy}{dt} \tag{12.404}$$

where breakup now occurs for $y>1$. For under-damped droplets, the equation governing y can easily be determined from Equation 12.404 (p. 557) if the relative velocity is assumed to be constant:

$$y(t)=We_c+e^{-(t/t_d)}\left[\left(y_0-We_c\right)\cos(\omega t)+\frac{1}{\omega}\left(\frac{dy_0}{dt}+\frac{y_0-We_c}{t_d}\right)\sin(\omega t)\right] \tag{12.405}$$

where

$$We=\frac{\rho_g u^2 r}{\sigma} \tag{12.406}$$

$$We_c=\frac{C_F}{C_k C_b}We \tag{12.407}$$

$$y_0=y(0) \tag{12.408}$$

$$\frac{dy_0}{dt}=\frac{dy}{dt}(0) \tag{12.409}$$

$$\frac{1}{t_d}=\frac{C_d}{2}\frac{\mu_l}{\rho_l r^2} \tag{12.410}$$

$$\omega^2=C_k\frac{\sigma}{\rho_l r^3}-\frac{1}{t_d^2} \tag{12.411}$$

In Equation 12.405 (p. 557), u is the relative velocity between the droplet and the gas phase and We is the droplet Weber number, a dimensionless parameter defined as the ratio of aerodynamic forces to surface tension forces. The droplet oscillation frequency is represented by ω. The default value of y_0 is 0, based upon the work of Liu et al. [382] (p. 1075). The constants have been chosen to match experiments and theory [329] (p. 1072):

$$C_k=8 \tag{12.412}$$

$$C_d=5$$

$$C_F=\frac{1}{3}$$

If Equation 12.405 (p. 557) is solved for all droplets, those with $y>1$ are assumed to break up. The size and velocity of the new child droplets must be determined.

12.13.1.4. Size of Child Droplets

The size of the child droplets is determined by equating the energy of the parent droplet to the combined energy of the child droplets. The energy of the parent droplet is [491] (p. 1081)

$$E_{parent}=4\pi r^2\sigma+K\frac{\pi}{5}\rho_l r^5\left[\left(\frac{dy}{dt}\right)^2+\omega^2y^2\right] \tag{12.413}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

where K is the ratio of the total energy in distortion and oscillation to the energy in the fundamental mode, of the order ($\frac{10}{3}$). The child droplets are assumed to be non-distorted and non-oscillating. Thus, the energy of the child droplets can be shown to be

$$E_{child}=4\pi r^2\sigma\frac{r}{r_{32}}+\frac{\pi}{6}\rho_l r^5\left(\frac{dy}{dt}\right)^2 \tag{12.414}$$

where r_{32} is the Sauter mean radius of the droplet size distribution. r_{32} can be found by equating the energy of the parent and child droplets (that is, Equation 12.413 (p. 557) and Equation 12.414 (p. 558)), setting $y=1$, and $\omega^2=8\sigma/\rho_l r^3$:

$$r_{32}=\frac{r}{1+\frac{8Ky^2}{20}+\frac{\rho_l r^3(dy/dt)^2}{\sigma}\left(\frac{6K-5}{120}\right)} \tag{12.415}$$

Once the size of the child droplets is determined, the number of child droplets can easily be determined by mass conservation.

12.13.1.5. Velocity of Child Droplets

The TAB model allows for a velocity component normal to the parent droplet velocity to be imposed upon the child droplets. When breakup occurs, the equator of the parent droplet is traveling at a velocity of $dx/dt=C_b r(dy/dt)$. Therefore, the child droplets will have a velocity normal to the parent droplet velocity given by

$$v_{normal}=C_v C_b r\frac{dy}{dt} \tag{12.416}$$

where C_v is a constant of order (1).

12.13.1.6. Droplet Breakup

To model droplet breakup, the TAB model first determines the amplitude for an undamped oscillation ($t_d\approx\infty$) for each droplet at time step n using the following:

$$A=\sqrt{\left(y^n-We_c\right)^2+\left(\frac{\left(dy/dt\right)^n}{\omega}\right)^2} \tag{12.417}$$

According to Equation 12.417 (p. 558), breakup is possible only if the following condition is satisfied:

$$We_c+A>1 \tag{12.418}$$

This is the limiting case, as damping will only reduce the chance of breakup. If a droplet fails the above criterion, breakup does not occur. The only additional calculations required then, are to update y using a discretized form of Equation 12.405 (p. 557) and its derivative, which are both based on work done by O'Rourke and Amsden [491] (p. 1081):

$$y^{n+1}=We_c+e^{-(\Delta t/t_d)}\left\{\left(y^n-We_c\right)\cos(\omega t)+\frac{1}{\omega}\left[\left(\frac{dy}{dt}\right)^n+\frac{y^n-We_c}{t_d}\right]\sin(\omega t)\right\} \tag{12.419}$$

$$\left(\frac{dy}{dt}\right)^{n+1}=\frac{We_c-y^{n+1}}{t_d}+\omega e^{-(\Delta t/t_d)}\left\{\frac{1}{\omega}\left[\left(\frac{dy}{dt}\right)^n+\frac{y^n-We_c}{t_d}\right]\cos(\omega\Delta t)-\left(y^n-We_c\right)\sin(\omega\Delta t)\right\} \tag{12.420}$$

All of the constants in these expressions are assumed to be constant throughout the time step.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

If the criterion of Equation 12.418 (p. 558) is met, then breakup is possible. The breakup time, t_{bu}, must be determined to see if breakup occurs within the time step Δt. The value of t_{bu} is set to the time required for oscillations to grow sufficiently large that the magnitude of the droplet distortion, y, is equal to unity. The breakup time is determined under the assumption that the droplet oscillation is undamped for its first period. The breakup time is therefore the smallest root greater than t^n of an undamped version of Equation 12.405 (p. 557):

$$We_c + A\cos\left[\omega\left(t - t^n\right) + \varphi\right] = 1 \tag{12.421}$$

where

$$\cos\varphi = \frac{y^n - We_c}{A} \tag{12.422}$$

and

$$\sin\varphi = -\frac{\left(dy/dt\right)^n}{A\omega} \tag{12.423}$$

If $t_{bu} > t^{n+1}$, then breakup will not occur during the current time step, and y and $\left(dy/dt\right)$ are updated by Equation 12.419 (p. 558) and Equation 12.420 (p. 558). The breakup calculation then continues with the next droplet. Conversely, if $t^n < t_{bu} < t^{n+1}$,then breakup will occur and the child droplet radii are determined by Equation 12.415 (p. 558). The number of child droplets, N, is determined by mass conservation:

$$N^{n+1} = N^n\left(\frac{r^n}{r^{n+1}}\right)^3 \tag{12.424}$$

It is assumed that the child droplets are neither distorted nor oscillating; that is, $y = \left(dy/dt\right) = 0$. The child droplets are represented by a number of child parcels which are created from the original parcel. These child parcels are distributed equally along the equator of the parent droplet in a plane normal to the parent relative velocity vector. The diameter of each of the child parcels is sampled from a Rosin Rammler distribution based on the Sauter mean radius (Equation 12.415 (p. 558)) and a spread parameter of 3.5.

A velocity component normal to the relative velocity vector, with magnitude computed by Equation 12.416 (p. 558), is imposed upon the child droplets. It is decomposed at the equator into components pointing radially outward.

Important:

A large number of child parcels ensures a smooth distribution of particle diameters and source terms which is needed when simulating, for example, evaporating sprays.

12.13.2. Wave Breakup Model

12.13.2.1. Introduction

An alternative to the TAB model that is appropriate for high-Weber-number flows is the wave breakup model of Reitz [544] (p. 1084), which considers the breakup of the droplets to be induced by the relative velocity between the gas and liquid phases. The model assumes that the time of breakup and the resulting droplet size are related to the fastest-growing Kelvin-Helmholtz instability,

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

derived from the jet stability analysis described below. The wavelength and growth rate of this instability are used to predict details of the newly-formed droplets.

12.13.2.2. Use and Limitations

The wave model is appropriate for high-speed injections, where the Kelvin-Helmholtz instability is believed to dominate droplet breakup ($We > 100$). Because this breakup model can increase the number of computational parcels, you may want to inject a modest number of droplets initially.

12.13.2.3. Jet Stability Analysis

The jet stability analysis described in detail by Reitz and Bracco [546] (p. 1084) is presented briefly here. The analysis considers the stability of a cylindrical, viscous, liquid jet of radius a issuing from a circular orifice at a velocity v into a stagnant, incompressible, inviscid gas of density ρ_2. The liquid has a density, ρ_1, and viscosity, μ_1, and a cylindrical polar coordinate system is used, which moves with the jet. An arbitrary infinitesimal axisymmetric surface displacement of the form

$$\eta = \eta_0 e^{ikz+\omega t} \tag{12.425}$$

is imposed on the initially steady motion and it is therefore desired to find the dispersion relation $\omega = \omega(k)$ which relates the real part of the growth rate, ω, to its wave number, $k = 2\pi / \lambda$.

In order to determine the dispersion relation, the linearized equations for the hydrodynamics of the liquid are solved assuming wave solutions of the form

$$\varphi_1 = C_1 I_0(kr) e^{ikz+\omega t} \tag{12.426}$$

$$\psi_1 = C_2 I_1(Lr) e^{ikz+\omega t} \tag{12.427}$$

where φ_1 and ψ_1 are the velocity potential and stream function, respectively, C_1 and C_2 are integration constants, I_0 and I_1 are modified Bessel functions of the first kind, $L^2 = k^2 + \omega / \nu_1$, and ν_1 is the liquid kinematic viscosity [544] (p. 1084). The liquid pressure is obtained from the inviscid part of the liquid equations. In addition, the inviscid gas equations can be solved to obtain the fluctuating gas pressure at $r = a$:

$$-p_{21} = -\rho_2 (U - i\omega k)^2 k\eta \frac{K_0(ka)}{K_1(ka)} \tag{12.428}$$

where K_0 and K_1 are modified Bessel functions of the second kind and u is the relative velocity between the liquid and the gas. The linearized boundary conditions are

$$v_1 = \frac{\partial \eta}{\partial t} \tag{12.429}$$

$$\frac{\partial u_1}{\partial r} = -\frac{\partial v_1}{\partial z} \tag{12.430}$$

and

$$-p_1 + 2\mu_1 - \frac{\sigma}{a^2}\left(\eta + a^2 \frac{\partial^2 \eta}{\partial z^2}\right) + p_2 = 0 \tag{12.431}$$

which are mathematical statements of the liquid kinematic free surface condition, continuity of shear stress, and continuity of normal stress, respectively. Note that u_1 is the axial perturbation liquid

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

velocity, v_1 is the radial perturbation liquid velocity, and σ is the surface tension. Also note that Equation 12.430 (p. 560) was obtained under the assumption that $v_2=0$.

As described by Reitz [544] (p. 1084), Equation 12.429 (p. 560) and Equation 12.430 (p. 560) can be used to eliminate the integration constants C_1 and C_2 in Equation 12.426 (p. 560) and Equation 12.427 (p. 560). Thus, when the pressure and velocity solutions are substituted into Equation 12.431 (p. 560), the desired dispersion relation is obtained:

$$\omega^2+2\nu_1 k^2\omega\left[\frac{I_1'(ka)}{I_0(ka)}-\frac{2kL}{k^2+L^2}\frac{I_1(ka)}{I_0(ka)}\frac{I_1'(La)}{I_1(La)}\right] = \frac{\sigma k}{\rho_1 a^2}\left(1-k^2a^2\right)\left(\frac{L^2-a^2}{L^2+a^2}\right)\frac{I_1(ka)}{I_0(ka)} + \frac{\rho_2}{\rho_1}\left(U-i\frac{\omega}{k}\right)^2\left(\frac{L^2-a^2}{L^2+a^2}\right)\frac{I_1(ka)}{I_0(ka)}\frac{K_0(ka)}{K_1(ka)} \tag{12.432}$$

As shown by Reitz [544] (p. 1084), Equation 12.432 (p. 561) predicts that a maximum growth rate (or most unstable wave) exists for a given set of flow conditions. Curve fits of numerical solutions to Equation 12.432 (p. 561) were generated for the maximum growth rate, Ω, and the corresponding wavelength, Λ, and are given by Reitz [544] (p. 1084):

$$\frac{\Lambda}{a}=9.02\frac{\left(1+0.45Oh^{0.5}\right)\left(1+0.4Ta^{0.7}\right)}{\left(1+0.87We_2^{1.67}\right)^{0.6}} \tag{12.433}$$

$$\Omega\sqrt{\frac{\rho_1 a^3}{\sigma}}=\frac{0.34+0.38We_2^{1.5}}{\left(1+Oh\right)\left(1+1.4Ta^{0.6}\right)} \tag{12.434}$$

where $Oh=\sqrt{We_1}/Re_1$ is the Ohnesorge number and $Ta=Oh\sqrt{We_2}$ is the Taylor number. Furthermore, $We_1=\rho_1 U^2 a/\sigma$ and $We_2=\rho_2 U^2 a/\sigma$ are the liquid and gas Weber numbers, respectively, and $Re_1=Ua/\nu_1$ is the Reynolds number.

12.13.2.4. Droplet Breakup

In the wave model, breakup of droplet parcels is calculated by assuming that the radius of the newly-formed droplets is proportional to the wavelength of the fastest-growing unstable surface wave on the parent droplet. In other words,

$$r=B_0\Lambda \tag{12.435}$$

where B_0 is a model constant set equal to 0.61 based on the work of Reitz [544] (p. 1084). Furthermore, the rate of change of droplet radius in the parent parcel is given by

$$\frac{da}{dt}=-\frac{\left(a-r\right)}{\tau},\ r\le a \tag{12.436}$$

where the breakup time, τ, is given by

$$\tau=\frac{3.726B_1 a}{\Lambda\Omega} \tag{12.437}$$

and Λ and Ω are obtained from Equation 12.433 (p. 561) and Equation 12.434 (p. 561), respectively. The breakup time constant, B_1, is set to a value of 1.73 as recommended by Liu et al. [382] (p. 1075). Values of B_1 can range between 1 and 60, depending on the injector characterization.

In the wave model, mass is accumulated from the parent drop at a rate given by Equation 12.437 (p. 561) until the shed mass is equal to 5% of the initial parcel mass. At this time, a new parcel is created with a radius given by Equation 12.435 (p. 561). The new parcel is given the same

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

properties as the parent parcel (that is, temperature, material, position, etc.) with the exception of radius and velocity. The new parcel is given a component of velocity randomly selected in the plane orthogonal to the direction vector of the parent parcel, and the momentum of the parent parcel is adjusted so that momentum is conserved. The velocity magnitude of the new parcel is the same as the parent parcel.

You must also specify the model constants that determine how the gas phase interacts with the liquid droplets. For example, the breakup time constant B1 is the constant multiplying the time scale that determines how quickly the parcel will lose mass. Therefore, a larger number means that it takes longer for the particle to lose a given amount. A larger number for B1 in the context of interaction with the gas phase would mean that the interaction with the subgrid is less intense. B0 is the constant for the drop size and is generally taken to be 0.61.

In the initial breakup of a cylindrical liquid jet at low gas Weber numbers (Rayleigh regime), Equation 12.435 (p. 561) can predict droplet diameters that are larger than the initial (jet) diameter. In that case, the size of the droplets is calculated from the fastest growing wavelength, which determines the spacing of the initial droplets, and mass conservation considerations. You can suppress this accounting for the Rayleigh regime as described in Breakup in the *Fluent User's Guide*.

12.13.3. KHRT Breakup Model

12.13.3.1. Introduction

Ansys Fluent provides an alternative model for droplet breakup called KHRT ([54] (p. 1056), [506] (p. 1082)), which combines the effects of Kelvin-Helmholtz waves driven by aerodynamic forces with Rayleigh-Taylor instabilities due to acceleration of shed drops ejected into freestream conditions. Both mechanisms model droplet breakup by tracking wave growth on the surface of the droplet, with breakup occurring due to the fastest growing instability based on local conditions. To model the liquid core in the near nozzle region, a Levich core length is applied that only allows breakup due to Kelvin-Helmholtz wave growth.

12.13.3.2. Use and Limitations

The KHRT model was developed to simulate high Weber number sprays and should not be applied to low pressure sprays.

12.13.3.3. Liquid Core Length

The KHRT model assumes that a liquid core exists in the near nozzle region. Child droplets are shed from this liquid core and are subjected to sudden acceleration when they are ejected into the freestream and a Rayleigh-Taylor instability becomes the dominant effect. The length of the liquid core is obtained from Levich theory [357] (p. 1073):

$$L=C_L d_0\sqrt{\frac{\rho_l}{\rho_g}} \tag{12.438}$$

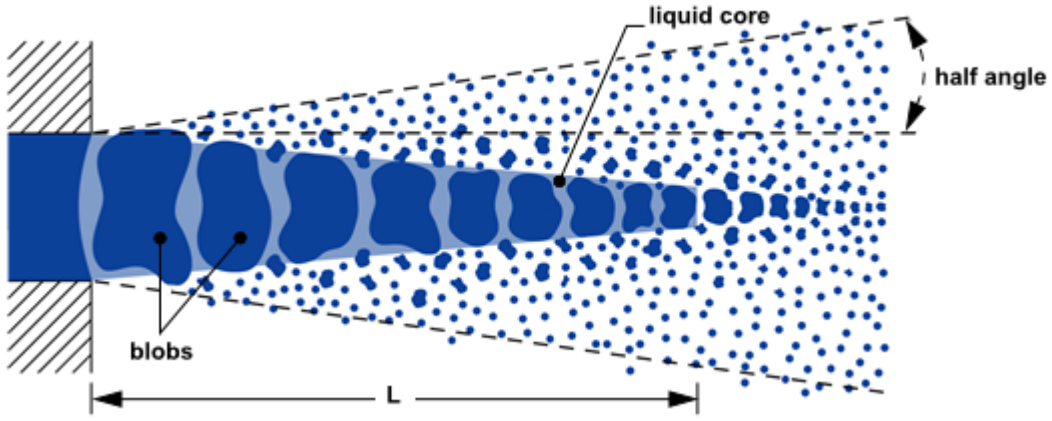
Where C_L is the Levich constant and d_0 is a reference nozzle diameter. The reference diameter must be specified for atomizer injections in Ansys Fluent, however the cone and solid cone injections assume that the reference diameter is the minimum of the droplet diameter and the inner diameter of the cone. Surface injections use the surface area specified to calculate an equivalent hydraulic

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

diameter for use as d_0. When a Rosin-Rammler distribution is specified for non-atomizer injections, the maximum diameter of the Rosin-Rammler becomes the reference diameter.

A liquid core is approximated by 'blobs' [54] (p. 1056) as shown in Figure 12.23: Liquid Core Approximation (p. 563).

Figure 12.23: Liquid Core Approximation



The 'blobs' are introduced into the domain with the diameter specified in the injection. It is often best to use an effective droplet diameter by considering a contraction coefficient, which can be computed with

$$D_e=\sqrt{C_a}\cdot d_0 \tag{12.439}$$

Where C_a denotes the contraction coefficient.

You can disable the Liquid Core Approximation entirely by setting the Levich constant, C_L, to zero as described in Breakup in the *Fluent User's Guide*.

12.13.3.4. Rayleigh-Taylor Breakup

As with the Kelvin-Helmholtz model, the Rayleigh-Taylor (RT) model is based on wave instabilities on the droplet surface. The frequency of the fastest growing wave is computed by

$$\Omega_{RT}=\sqrt{\frac{2(-g_t(\rho_p-\rho_g))^{3/2}}{3\sqrt{3\sigma}\,(\rho_p+\rho_g)}} \tag{12.440}$$

Where g_t is the droplet acceleration in the direction of the droplet travel. The corresponding wave number is given by

$$K_{RT}=\sqrt{\frac{-g_t(\rho_p-\rho_g)}{3\sigma}} \tag{12.441}$$

Breakup occurs after RT waves have been growing for a time larger than the breakup time τ_{RT}, defined as

$$\tau_{RT}=\frac{C_\tau}{\Omega_{RT}} \tag{12.442}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Here C_τ is the Rayleigh-Taylor breakup time constant which has a default value of 0.5.

The wave growth is tracked when the predicted wave length $2\pi C_{RT}/K_{RT}$, corresponding to the fastest wave growth rate, is smaller than the local droplet diameter. The radius of the smaller child droplets is calculated with

$$r_c = \frac{\pi C_{RT}}{K_{RT}} \tag{12.443}$$

where C_{RT} is the breakup radius constant which has a default value of 0.1.

12.13.3.5. Droplet Breakup Within the Liquid Core

Within the liquid core only aerodynamic breakup is considered. It is computed based on the Wave breakup model, described in the previous section.

12.13.3.6. Droplet Breakup Outside the Liquid Core

Outside the liquid core, both KH and RT effects are calculated and both are considered for breakup. Typically, the RT instability grows faster when droplet acceleration is high and this effect dominates for high Weber number sprays. In both cases, a new parcel is generated only when the accumulated shed parcel mass exceeds 5% of the initial parcel mass.

12.13.4. Stochastic Secondary Droplet (SSD) Model

In the **Wave** and **TAB** spray models, droplet breakup is determined using a single diameter scale. The **SSD** model [24] (p. 1054) treats breakup as a discrete random event resulting in a distribution of diameter scales over a range. With the **SSD** model, the probability of breakup is independent of the parent droplet size and the secondary droplet size is sampled from an analytical solution of the Fokker-Planck equation for the probability distribution. In addition, parameters for the size distribution are based on local conditions.

The initial diameter of the parcels injected into the domain is set in the DPM injection dialog box. The breakup model predicts the time at which breakup occurs and the number and properties of the new drops. Drops larger than a critical radius are subject to breakup:

$$r_c = \frac{We_{cr}\sigma_l}{\rho_g u_{rel}^2} \tag{12.444}$$

where We_{cr} is the critical Weber number, which you will need to specify. The default value of the critical Weber number is 6.

The breakup time is defined as

$$t_{bu} = B\sqrt{\frac{\rho_l}{\rho_g}}\frac{r}{|u_{rel}|} \tag{12.445}$$

where B is the user-specified breakup constant, with a default value of 1.73.

Drops with a radius larger than the critical radius (derived from the critical Weber number) have their breakup times incremented. When the breakup time on the parcel is larger than the critical breakup time (locally calculated from conditions in the cell and on the parcel), breakup occurs.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

When a parcel reaches breakup, it is destroyed and new parcels are created. The diameters of these child parcels are obtained by sampling a distribution function in the log of the diameter, $x = \ln(r)$:

$$T(x) = \frac{1}{\sqrt{2\pi\langle\xi^2\rangle}} \exp\left[\frac{-(x - x_0 - \langle\xi\rangle)^2}{2\langle\xi^2\rangle}\right] \tag{12.446}$$

where $\langle\xi\rangle$ and $\langle\xi^2\rangle$ are parameters of the model.

The parameter $\langle\xi\rangle$ has a negative dimensionless value and must be set by you: $\exp(\langle\xi\rangle)$ is a typical factor by which child particles are smaller than the original parcel.

The parameter $\langle\xi^2\rangle$ defines the variance, in logarithmic terms, of the distribution of child particles. It has a positive, dimensionless value, which should be significantly less than $\langle\xi\rangle^2$ (to avoid child particles that are larger than the original). It is computed from $\langle\xi^2\rangle = -0.1\ln\left(\frac{We_{cr}}{We}\right)$.

When breakup occurs, enough parcels are created so that the number of drops represented by each parcel is approximately equal to a target number in the parcel (NP), which you set. Parcels are created, with diameters sampled randomly from the distribution, and provisionally given the target NP. This continues until the mass of the parent parcel is used up. A scaling factor is then applied to the number of drops in all the new parcels to conserve the mass of the parent parcel.

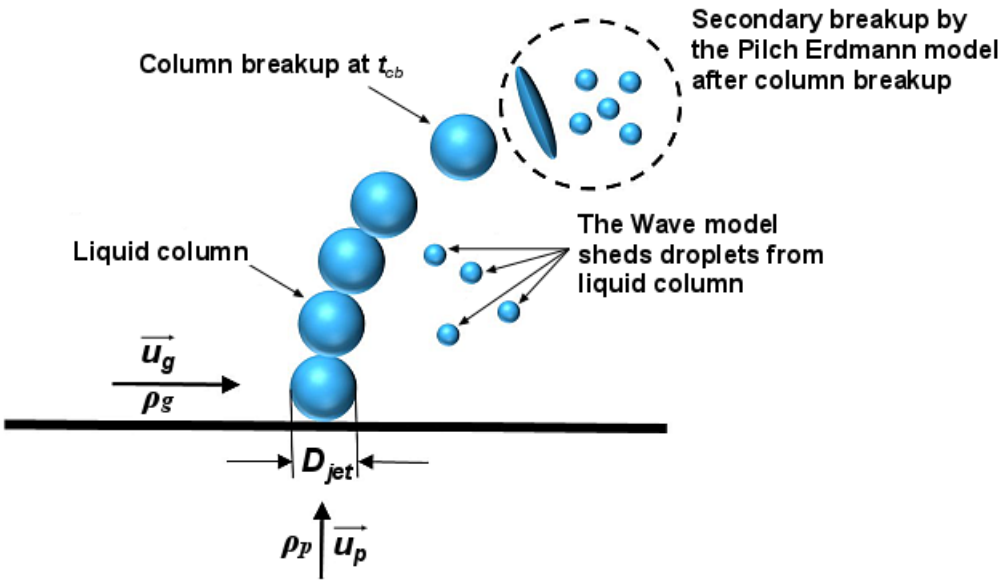
This methodology results in improved statistics and gives you control over error during a simulation. A smaller NP will mean more parcels but lower statistical error. However, there is a limit to the number of parcels that can be created during a single breakup event; the default value for this limit is 50. Therefore, the number of drops in each parcel will sometimes deviate significantly from the target NP.

12.13.5. Madabhushi Breakup Model

In the Madabhushi framework [404] (p. 1076), the effects of primary breakup are simulated by the Wave model, and the effects of secondary breakup are simulated by a model suggested by Pilch and Erdman [514] (p. 1083). Figure 12.24: Madabhushi Breakup Model (p. 566) shows particle breakup mechanisms considered in the Madabhushi breakup model.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Figure 12.24: Madabhushi Breakup Model



The droplets are injected through an orifice with an exit diameter D_{jet}. In the primary breakup stage, the jet is represented by a liquid column of spherical droplets of equal diameters D_0=D_{jet}. The initial droplet velocity u_p is calculated from the inlet jet velocity and liquid density (or the flow rate). During this phase, child droplets are shed according to the standard WAVE breakup model. Subsequent breakups (secondary breakup due to turbulence, capillary and aerodynamic forces) for such child droplets are modeled using the Pilch and Erdman model [514] (p. 1083).

The droplet stays in the liquid column until the so-called column breakup time t_{cb} is reached:

$$t_{cb}=C_0\frac{D_0}{u_g}\sqrt{\frac{\rho_l}{\rho_g}}, \quad C_0=3.44 \tag{12.447}$$

where u_g is the gas velocity of the crossflow, ρ_l the liquid density, and ρ_g the gas density. C_0 is the column breakup time constant. Its value varies in different publications. Ansys Fluent uses the value proposed in the work of Madabhushi [404] (p. 1076) C_0 = 3.44. During this phase, the drag remains constant C_D = 1.48.

Once the droplet lifetime exceeds the column breakup time t_{cb}, it undergoes secondary breakup according to Pilch and Erdman's model [514] (p. 1083). In this model, the droplet is deformed from a sphere to a disk within a deformation period t_{def}:

$$t_{def}=1.6\ t^* \tag{12.448}$$

where t^* is the characteristic time scale for breakup given by:

$$t^*=\frac{D_{parent}}{u_{rel}}\sqrt{\frac{\rho_l}{\rho_g}} \tag{12.449}$$

Here, u_{rel} is the relative velocity between droplet and gas, and D_{parent} is the local droplet diameter at the column breakup time t_{cb}.

During the deformation period, the droplet's drag changes according to the following expression:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$C_D = C_D^{sphere}\left(1-\frac{t}{t_{def}}\right) + C_D^{disc}\frac{t}{t_{def}}\ ,\ t < t_{def} \tag{12.450}$$

where the drag coefficient of the disc C_D^{disc} =1.2.

The reference area for drag changes according to the following expression:

$$D_{ref} = \begin{cases} \left(1+0.19\sqrt{We}\,\frac{t}{t_{def}}\right)D_{parent} & t<t_{def},\ We<100 \\ \left(1+1.9\frac{t}{t_{def}}\right)D_{parent} & t<t_{def},\ We\geq 100 \end{cases} \tag{12.451}$$

The drag force acting on the droplet changes as:

$$F_D = \frac{1}{2}\rho_g C_D A_P^* \cdot \left|\vec{u}_g - \vec{u}_p\right| \cdot (\vec{u}_g - \vec{u}_p) \tag{12.452}$$

where C_D is the drag coefficient and A_P^* is the cross section area calculated by:

$$A_P^* = \frac{\pi}{4}D_{ref}^2 = \frac{D_{ref}^2}{D_{parent}^2}\frac{\pi}{4}D_{parent}^2 = \left(\frac{D_{ref}}{D_{parent}}\right)^2 A_P^{parent} \tag{12.453}$$

Once the deformation time t_{def} is reached, the droplet remains deformed and the drag stays constant:

$$C_D = C_D^{disc},\ D_{ref} = \begin{cases} (1+0.19\sqrt{We}\,)D_{parent} & We<100 \\ 2.9D_{parent} & We\geq 100 \end{cases} \tag{12.454}$$

until the droplet breaks up into five smaller child droplet parcels.

The total breakup time t_b is dependent on the local Weber number and is determined by the following correlations:

$$\frac{t_b}{t^*} = \begin{cases} 6(We-12)^{-0.25}, & 12< We <18 \\ 2.45(We-12)^{0.25}, & 18< We <45 \\ 14.1(We-12)^{-0.25}, & 45< We <351 \\ 0.776(We-12)^{0.25}, & 351 < We < 2670 \\ 5.5, & 2670 < We \end{cases} \tag{12.455}$$

Due to the rim expansion, the velocity of each child droplet $\vec{u}_{child}$ is calculated as:

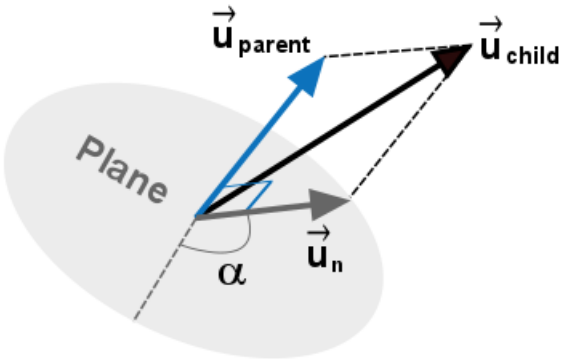
$$\vec{u}_{child} = \vec{u}_{parent} + \vec{u}_n \tag{12.456}$$

where $\vec{u}_{parent}$ is the parent droplet velocity, and $\vec{u}_n$ is a normal velocity in a plane normal to the parent droplet velocity $\vec{u}_{parent}$ (see Figure 12.25: Child Droplet Velocity (p. 568)); its magnitude is expressed as:

$$u_n = 5\frac{D_{parent}}{t_b - t_{def}} \tag{12.457}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Figure 12.25: Child Droplet Velocity



For each of the five droplets, the normal velocity direction angle α is randomly chosen in the range [0, 2π].

The target volumetric distribution of child droplets after breakup is given by the following root-normal distribution:

$$f\left(D_{child}\right)=\frac{\sqrt{\frac{D_{child}}{D_{0.5}}}}{2\sqrt{2\pi}\ \ 0.238\ D_{child}}exp\left(-\frac{1}{2}\left(\frac{\sqrt{\frac{D_{child}}{D_{0.5}}}-1}{0.238}\right)^{2}\right) \tag{12.458}$$

where $D_{0.5}=\ 1.2\ D_{SMD}$ and denotes the mass median diameter related to the target Sauter Mean Diameter after breakup as:

$$D_{SMD}=1.5\frac{Oh^{0.2}}{We_{corr}^{0.25}}\ D_{parent} \tag{12.459}$$

where Oh is the Ohnesorge number given by:

$$Oh=\frac{\mu_l}{\sqrt{\rho_l\ D_{ref}\ \sigma_l}} \tag{12.460}$$

Here, μ_l is the droplet viscosity, σ_l is the droplet surface tension, and D_{ref} is the reference area of the deformed droplet given by:

$$D_{ref}=\begin{cases}\left(1+0.19\sqrt{We}\right)D_{\text{parent}}, & We\ <\ 100\\ 2.9\ D_{\text{parent}}, & We\geq\ 100\end{cases} \tag{12.461}$$

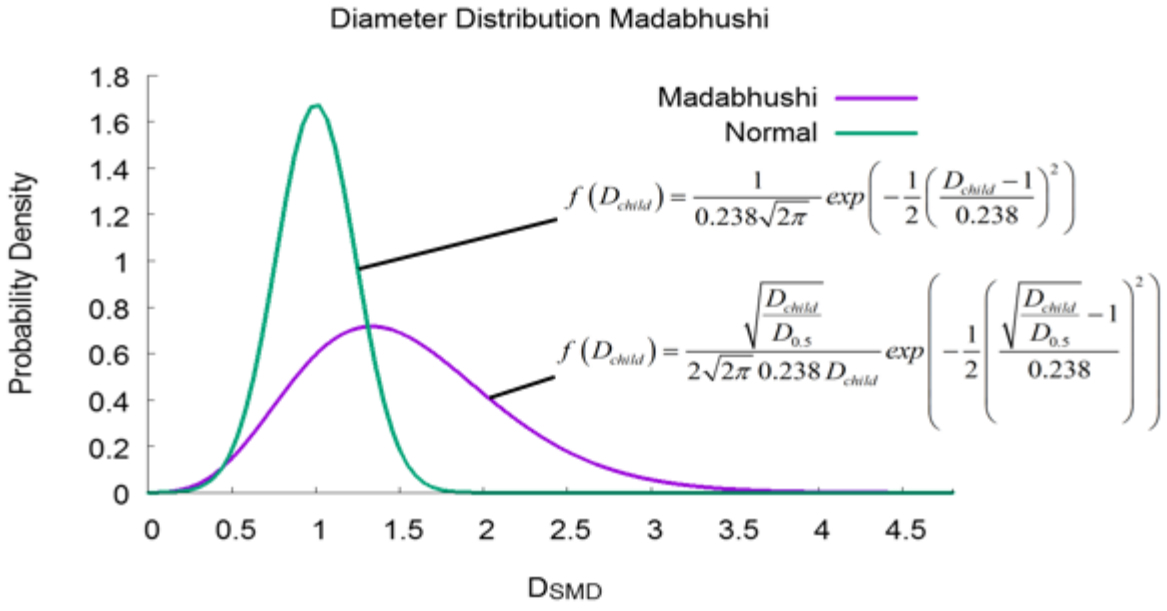
where We_{corr} is the corrected Weber number for high droplet viscosity (Oh > 0,1); it is calculated as:

$$We_{corr}=\frac{We}{1+1.077\ Oh^{1.6}} \tag{12.462}$$

The Madabhushi diameter distribution of the child droplets after breakup compared with the standard normal distribution is shown in Figure 12.26: Madabhushi Diameter Distribution (p. 569).

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Figure 12.26: Madabhushi Diameter Distribution



The droplets that undergo secondary breakup after column breakup represent real-world ligaments breaking up from the liquid core. The ligaments vary in shape and eventually break into smaller ligaments that are not equal in size to the original child droplets produced by the Pilch and Erdman model, which assumes almost spherical parent droplets. The Pilch and Erdman breakup model have the tendency to overestimate the child droplet diameters in this region. To overcome this inaccuracy, the diameters of the child droplets that undergo further secondary breakup after column breakup are weighted by a factor F_{Lig}>0 to consider the influence of ligaments:

$$D_{child}=F_{Lig}\overline{D}_{child} \tag{12.463}$$

$\overline{D}_{child}$ follows the same target volumetric distribution of child droplets after breakup shown in Equation 12.458 (p. 568).

In this mechanism, the child droplets continue to break up further according to Pilch and Erdman's model with no child droplet diameter weighting (that is, F_{Lig}=1) until they become so small that the surface tension of the water begins to form new droplets and no further breakup occurs.

The same mechanism (that is F_{Lig}=1) is applied to further secondary breakup of droplets shed from the liquid core by the WAVE breakup model.

12.13.6. Schmehl Breakup Model

The Schmehl breakup model [577] (p. 1086) distinguishes between three breakup regimes:

- Bag Breakup
- Multimode Breakup
- Shear Breakup

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

These regimes differ by the Weber number, which is a measure of strength of aerodynamic forces relative to surface tension forces:

$$We=\frac{\rho_g u_{rel}^2 D_{parent}}{\sigma_l} \tag{12.464}$$

and the Ohnesorge number, which assesses the damping effect of viscous friction in the droplet against surface tension:

$$Oh=\frac{\mu_l}{\sqrt{\rho_l D_{parent}\sigma_l}} \tag{12.465}$$

In Equation 12.464 (p. 570) and Equation 12.465 (p. 570):

ρ_g = gas density

u_{rel} = relative droplet-gas-velocity

D_{parent} = diameter of the droplet before breakup

ρ_l = droplet liquid density

σ_l = surface tension of droplet liquid

The transition between different breakup mechanisms are modeled by the following functions depending on the local Weber number:

- Deformation with no breakup:

$$\mathrm{We} \le We_{crit}=12\left(1+1.077Oh^{1.6}\right)$$

- Bag Breakup:

$$We_{crit} < \mathrm{We} \le We_{multimode}=20\left(1+1.200Oh^{1.5}\right)$$

- Multimode Breakup:

$$We_{multimode} < \mathrm{We} \le We_{shear}=32\left(1+1.500Oh^{1.4}\right)$$

- Shear Breakup:

$$We_{shear} < \mathrm{We}$$

The three breakup mechanisms use different normal velocities, breakup time-points, and volume distribution of child droplets.

In all regimes, the breakup process is subdivided into two stages:

- Initial deformation of the droplet into a disc shape
- Further deformation with disintegration of the droplets

The Schmehl breakup model uses the deformation and disintegration mechanism suggested by Pilch and Erdman [515] (p. 1083). The droplet deformation period, drag coefficient, reference area for drag,

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

and drag force are computed according to Equation 12.448 (p. 566) through Equation 12.454 (p. 567) (see Madabhushi Breakup Model (p. 565) for details).

The total breakup time t_b is dependent on the local Weber number and is determined by the following correlations:

- For Ohnesorge numbers $Oh < 1$, Equation 12.455 (p. 567) is used.
- For Ohnesorge numbers $Oh \geq 1$, liquid viscosity is dominating the droplet breakup process, and the following correlation is used:

$$\frac{t_b}{t^*} = 4.5\left(1 + 1.2Oh^{0.74}\right) \tag{12.466}$$

The child droplet velocity due to the rim expansion is calculated by Equation 12.456 (p. 567) (see Figure 12.25: Child Droplet Velocity (p. 568)). The magnitude of the normal velocity $\vec{u}_n$ in a plane normal to the parent droplet velocity $\vec{u}_{parent}$ is dependent on the breakup regime:

- Bag Breakup ($We_{crit} < \mathrm{We} \leq We_{multimode}$):

$$u_n = \frac{2.5\ D_{parent}}{t_b - t_{def}} \tag{12.467}$$

- Multimode Breakup ($We_{mulimode} < \mathrm{We} \leq We_{shear}$):

$$u_n = \frac{2.0\ D_{parent}}{t_b - t_{def}} \tag{12.468}$$

- Shear Breakup ($We_{shear} < We$):

$$u_n = 0 \tag{12.469}$$

Like for the Madabhushi model, the normal velocity direction angle α is randomly chosen in the range [0, 2π] for each of the child droplets (see Figure 12.25: Child Droplet Velocity (p. 568)). The number of child droplets is not fixed to five as in the Madabhushi model. By default, it is set to 1, but can be changed in the GUI as described in Breakup in the *Fluent User's Guide*.

In the Bag Breakup and Multimode Breakup regimes, the target volumetric distribution of child droplets after breakup is given by the root-normal distribution Equation 12.458 (p. 568) known from the Madabhushi breakup model (see Figure 12.26: Madabhushi Diameter Distribution (p. 569)).

In the Bag Breakup regime, the total mass of the parent droplet parcel M_{parent} is uniformly distributed over the child droplets parcels. In the Shear Breakup regime, 20 % of the total mass of the parent droplet parcel M_{parent} remains in the parent droplet parcel, and only 80 % is uniformly distributed over the child droplets parcels. The Multimode Breakup regime is in between the Bag Breakup and the Shear Breakup regimes. This regime computes the shed mass using a linear interpolation between the two other regimes based on the Weber numbers that define the breakup regimes $We_{mulimode}$ and We_{shear}.

In the Multimode Breakup regime ($We_{mulimode} < \mathrm{We} \leq We_{shear}$), the mass remaining in the parent droplet parcel M_{core} is calculated as:

$$M_{core} = 0.2\ \frac{We - We_{multimode}}{We_{shear} - We_{multimode}} M_{parent} \tag{12.470}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Accordingly, the mass uniformly distributed over the child droplets parcels $M_{stripped}$ is computed as:

$$M_{stripped}=\left(1\text{-}0.2\ \frac{We\text{-}We_{multimode}}{We_{shear}\text{-}We_{multimode}}\right)M_{parent} \tag{12.471}$$

Finally, the remaining parent droplet parcel is adapted based on the new diameter following the target volumetric distribution of its child droplets after breakup and also the new flow rate and the number in parcel, which are calculated based on M_{core}.

In the Shear Breakup regime, the volume distribution of child droplets is characterized by a bimodal density function with a maximum in the region of small droplet diameters and a second maximum in the region of large droplet diameters. The fine fraction of the droplet fragments represents approximately 80% of the total mass of the parent droplet parcel M_{parent}. These droplets result from film fragments stripped off the disc-shaped droplet by shear forces. The remaining 20% in the large diameter region represents the contribution of the core droplet fragments formed by the stripping process. The diameter D_{crit} of this core droplet is estimated from the critical Weber number We_{crit} at flow conditions at the instant of breakup as:

$$D_{crit}=\frac{We_{crit}\ \sigma_l}{\rho_g\ u_{rel}^2\ D_{parent}} \tag{12.472}$$

The target volumetric distribution of the stripped child droplets after breakup is given by the same root-normal distribution as for the Madabhushi breakup model Equation 12.458 (p. 568) (see Figure 12.26: Madabhushi Diameter Distribution (p. 569)); however, here, the mass media diameter is taken as:

- For $D_{crit}>D_{SMD}$,

$$D_{0.5}=1.2\ D_{SMD}^{reduced} \tag{12.473}$$

  with

$$D_{SMD}^{reduced}=\frac{4\ D_{SMD}\ D_{crit}}{5D_{crit}\text{-}D_{SMD}} \tag{12.474}$$

- For $D_{crit}\leq D_{SMD}$,

$$D_{0.5}=1.2\ D_{SMD} \tag{12.475}$$

  as in the Bag Breakup and Multimode Breakup regimes.

The remaining parent droplet parcel is then adapted according to the new diameter:

$$D_{parent}^{new}=\begin{cases}max\left(D_{crit},D_{parent}\right) & if\ D_{crit}>D_{SMD}\\ D_{0.5} & if\ D_{crit}\leq D_{SMD}\end{cases} \tag{12.476}$$

and also the new flow rate and the number in parcel, which are calculated based on $M_{core}=0.2\ M_{parent}$. Since D_{crit} may be lager than the original parent droplet diameter depending on the critical Weber number We_{crit} , the maximum diameter of the remaining droplet parcel is restricted to the original parent droplet parcel diameter before the breakup process.

If the number of breakup parcels is set to one, a statistical breakup is complete, and no child droplet parcels are shed. The parent droplet parcel is adapted according to the new diameter, the new flow rate, and the number in parcel as follows:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

- Deformation (We $\leq We_{crit}$): No breakup
- Bag Breakup (We_{crit}<We$\leq We_{multimode}$):

$$D_{parent}^{new}=D_{0.5}=1.2\ D_{SMD} \tag{12.477}$$

- Multimode Breakup ($We_{mulimode}$<We$\leq We_{shear}$):

$$D_{parent}^{new}=D_{0.5}=1.2\ D_{SMD} \tag{12.478}$$

- Shear Breakup (We_{shear}<We):

$$D_{parent}^{new}=\begin{cases}1.2\ D_{SMD}^{reduced}=\frac{4\ D_{SMD}\ D_{crit}}{5D_{crit}-D_{SMD}} & if\ D_{crit}>D_{SMD}\\ D_{0.5}=1.2\ D_{SMD} & if\ D_{crit}\leq D_{SMD}\end{cases} \tag{12.479}$$

The new parent droplet diameter relates to the target Sauter mean diameter after breakup as:

$$D_{SMD}=1.5\frac{Oh^{0.2}}{We_{corr}^{0.25}}D_{parent} \tag{12.480}$$

The deformation and disintegration mechanism as well as the velocity adjustment due to the rim expansion of the parent droplet parcel is applied as described above.

12.14. Collision and Droplet Coalescence Model Theory

For information about this model, see the following sections:

12.14.1. Introduction

12.14.2. Use and Limitations

12.14.3. Theory

12.14.1. Introduction

When your simulation includes tracking of particles, Ansys Fluent provides an option for estimating the number of particle collisions and their outcomes in a computationally efficient manner. The difficulty in any collision calculation is that for N particles, each particle has $N-1$ possible collision partners. Thus, the number of possible collision pairs is approximately $\frac{1}{2}N^2$. (The factor of $\frac{1}{2}$ appears because particle A colliding with particle B is identical to particle B colliding with particle A. This symmetry reduces the number of possible collision events by half.)

An important consideration is that the collision algorithm must calculate $\frac{1}{2}N^2$ possible collision events at every time step. Since a spray can consist of several million particles, the computational cost of a collision calculation from first principles is prohibitive. This motivates the concept of parcels. Parcels are statistical representations of a number of individual particles. For example, if Ansys Fluent tracks a set of parcels, each of which represents 1000 particles, the cost of the collision calculation is reduced by a factor of 10^6. Because the cost of the collision calculation still scales with the square of N, the reduction of cost is significant; however, the effort to calculate the possible intersection of so many parcel trajectories would still be prohibitively expensive.

The algorithm of O'Rourke [490] (p. 1081) efficiently reduces the computational cost of the spray calculation. Rather than using geometry to see if parcel paths intersect, O'Rourke's method is a stochastic

estimate of collisions. O'Rourke also assumes that two parcels may collide only if they are in the same continuous-phase cell. These two assumptions are valid only when the continuous-phase cell size is small compared to the size of the spray. For these conditions, the method of O'Rourke is second-order accurate at estimating the chance of collisions. The concept of parcels together with the algorithm of O'Rourke makes the calculation of collision possible for practical spray problems.

Once it is decided that two parcels collide, the algorithm further determines the type of collision. For liquid droplets, coalescence and bouncing outcomes are considered, while for solid particles, only bouncing is considered. The probability of an outcome is calculated from the collisional Weber number (We_c) and a fit to experimental observations. Here,

$$We_c = \frac{\rho U_{rel}^2 \overline{D}}{\sigma} \tag{12.481}$$

where U_{rel} is the relative velocity between two parcels, and $\overline{D}$ is the arithmetic mean diameter of the particles represented by the two parcels. The state of the two colliding parcels is modified based on the outcome of the collision.

12.14.2. Use and Limitations

The collision model assumes that the frequency of collisions is much less than the particle time step. If the particle time step is too large, then the results may be time-step-dependent. You should adjust the particle length scale accordingly. Additionally, the model is most applicable for low-Weber-number collisions where collisions result in bouncing and coalescence. Above a Weber number of about 100, the outcome of collision could be shattering.

Sometimes the collision model can cause mesh-dependent artifacts to appear in the spray. This is a result of the assumption that parcels can collide only if they are within the same cell. These artefacts tend to be visible when the source of injection is at a mesh vertex. The coalescence of parcels tends to cause the spray to pull away from cell boundaries. In two dimensions, a finer mesh and more parcels can be used to reduce these effects. In three dimensions, best results are achieved when the spray is modeled using a polar mesh with the spray at the center.

If the collision model is used in a transient simulation, multiple DPM iterations per time step cannot be specified in the **DPM Iteration Interval** field in the Discrete Phase Model Dialog Box. In such cases, only one DPM iteration per time step will be calculated.

12.14.3. Theory

As noted above, O'Rourke's algorithm assumes that two parcels may collide only if they are in the same continuous-phase cell. This assumption can prevent parcels that are quite close to each other, but not in the same cell, from colliding, although the effect of this error is lessened by allowing some parcels that are farther apart to collide. The overall accuracy of the scheme is second-order in space.

12.14.3.1. Probability of Collision

The probability of collision of two particles is derived from the point of view of the larger particle, called the collector particle and identified below with the number 1. The smaller particle is identified in the following derivation with the number 2. The collision calculation is performed in the reference frame of the collector particle.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

In this local reference frame, a collision can occur only if the distance between the two centers of particle 1 and particle 2 in the normal direction to their relative velocity vector v_{rel} is less than or equal to r_1+r_2. Using this geometrical condition, a cross-section can be determined through which the center of particle 2 must pass in order for a collision to occur. The area of this cross-section is $\pi\left(r_1+r_2\right)^2$. The so-called collision volume can be derived by multiplying the cross-sectional area by the relative velocity vector v_{rel} and the integration time step Δt.

The algorithm of O'Rourke uses the collision volume to calculate the probability of a collision. Rather than calculating whether the center of particle 2 is within the collision volume, the algorithm calculates the probability of this event. If there is a uniform probability of the particle 2 being anywhere within a cell of volume V, then the chance of this particle being within the collision volume is the ratio of the collision volume to the cell volume. Thus, the probability of the collector colliding with particle 2 is:

$$P_1=\frac{\pi\left(r_1+r_2\right)^2 v_{rel}\Delta t}{V} \tag{12.482}$$

Equation 12.482 (p. 575) can be generalized for parcels, where there are n_1 particles in parcel 1 and n_2 particles in parcel 2. The collector parcel 1 undergoes a mean expected number of collisions given by:

$$\overline{n}=\frac{n_2\pi\left(r_1+r_2\right)^2 v_{\mathrm{rel}}\Delta t}{V} \tag{12.483}$$

The actual number of collisions that the collector experiences is not generally the mean expected number of collisions. The probability distribution of the number of collisions follows a Poisson distribution, according to O'Rourke, which is given by:

$$P(n)=e^{-\overline{n}}\frac{\overline{n}^n}{n!} \tag{12.484}$$

where n is the number of collisions between a collector and other particles.

For each DPM timestep, the mean expected number of collisions is calculated for each pair of tracked parcels in each cell. To determine whether a collision will occur, the probability of no collision ($P(0)=e^{-\overline{n}}$) is computed and compared to a uniformly sampled random number. If the random number is larger than $P(0)$, the two parcels will collide.

12.14.3.2. Collision Outcomes

Once it is determined that the two parcels collide, the outcome of the collision must be decided. In general, the following collision outcomes are possible:

- Bouncing only

  Bouncing occurs for any type of collision if coalescence effects are neglected.

- Bouncing and coalescence

  Coalescence may typically occur for liquid droplets.

  Coalescence happens if the particles collide head-on. If the collision is more oblique, the particles bounce off each other.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

12.14.3.2.1. Bouncing

If coalescence is not considered, the outcome of particle collision is a central collision. The collision point at which the two colliding particles touch is determined randomly as a point of intersection of the line connecting the centroids of the two colliding particles and their surfaces. In the frame of reference of the collector particle, the collision direction is described by the basis vector pointing in the direction of the line connecting the collector particle centroid and the collision point. Therefore, derived from the laws of momentum and energy conservation, the post-collisional velocity of the particles in $\mathbf{n}$ direction is computed by the following equation [221] (p. 1065):

$$v_1' = \frac{m_1 v_{1,\mathbf{n}} + m_2 v_{2,\mathbf{n}}}{m_1 + m_2} - \frac{k m_2 (v_{1,\mathbf{n}} - v_{2,\mathbf{n}})}{m_1 + m_2} \tag{12.485}$$

where m_1 and m_2 are the masses of particles 1 and 2, respectively, and k is the restitution coefficient. $k = 1$ in ideal collisions, while $k = 0$ in plastic collisions. Ansys Fluent assumes perfectly elastic collisions and uses a coefficient of restitution $k = 1$. Note that only the velocities in the $\mathbf{n}$ direction are changed.

12.14.3.2.2. Bouncing and Coalescence

In the reference frame being used here, the probability of coalescence can be related to the offset of the collector particle center and the trajectory of the smaller particle 2. The critical offset is a function of the collisional Weber number and the relative radii of the collector and the smaller particle.

The critical offset is calculated by O'Rourke using the expression

$$b_{crit} = (r_1 + r_2)\sqrt{\min\left(1.0, \frac{2.4f}{We}\right)} \tag{12.486}$$

where f is a function of r_1 / r_2, defined as

$$f\left(\frac{r_1}{r_2}\right) = \left(\frac{r_1}{r_2}\right)^3 - 2.4\left(\frac{r_1}{r_2}\right)^2 + 2.7\left(\frac{r_1}{r_2}\right) \tag{12.487}$$

The value of the actual collision parameter b is $(r_1 + r_2)\sqrt{Y}$, where Y is a random number between 0 and 1. The calculated value of b is compared to b_{crit}, and if $b < b_{crit}$, the result of the collision is coalescence. Equation 12.484 (p. 575) gives the number of smaller particles that coalesce with the collector. The properties of the coalesced particles are found from the basic conservation laws.

In the case of a grazing collision, the new velocities are calculated based on conservation of momentum and kinetic energy. It is assumed that some fraction of the kinetic energy of the particles is lost to viscous dissipation and angular momentum generation. This fraction is related to b, the collision offset parameter. Using assumed forms for the energy loss, O'Rourke derived the following expression for the new velocity:

$$v_1' = \frac{m_1 v_1 + m_2 v_2}{m_1 + m_2} + \frac{m_2 (v_1 - v_2)}{m_1 + m_2}\left(\frac{b - b_{crit}}{r_1 + r_2 - b_{crit}}\right) \tag{12.488}$$

This relation is used for each of the components of velocity. No other particle properties are altered in grazing collisions.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

12.15. Discrete Element Method Collision Model

Ansys Fluent allows you to use the discrete element method (DEM) as part of the DPM capability. To use the DEM model, refer to Modeling Collision Using the DEM Model in the User's Guide.

The discrete element method is suitable for simulating granular matter (such as gravel, coal, beads of any material). Such simulations are characterized by a high volume fraction of particles, where particle-particle interaction is important. Note that interaction with the fluid flow may or may not be important. Typical applications include:

- hoppers
- risers
- packed beds
- fluidized beds
- pneumatic transport

12.15.1. Theory

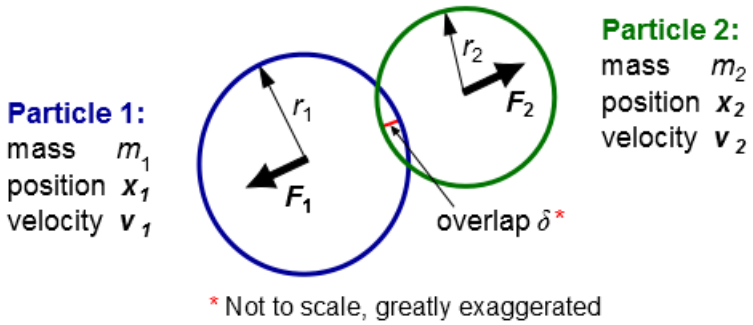
The DPM capabilities allow you to simulate moving particles as moving mass points, where abstractions are used for the shape and volume of the particles. Note that the details of the flow around the particles (for example, vortex shedding, flow separation, boundary layers) are neglected. Using Newton's second law, the ordinary differential equations that govern the particle motion are represented as follows:

$$m\frac{d\vec{v}}{dt} = \vec{F}_{\text{drag}} + \vec{F}_{\text{pressure}} + \vec{F}_{\text{virtual_mass}} + \vec{F}_{\text{gravitation}} + \vec{F}_{\text{other}} \tag{12.489}$$

$$\frac{dx}{dt} = \vec{v} \tag{12.490}$$

The DEM implementation is based on the work of Cundall and Strack [126] (p. 1060), and accounts for the forces that result from the collision of particles (the so-called "soft sphere" approach). These forces then enter through the $\vec{F}_{\text{other}}$ term in Equation 12.489 (p. 577). The forces from the particle collisions are determined by the deformation, which is measured as the overlap between pairs of spheres (see Figure 12.27: Particles Represented by Spheres (p. 577)) or between a sphere and a boundary. Equation 12.489 (p. 577) is integrated over time to capture the interaction of the particles, using a time scale for the integration that is determined by the rigidity of the materials.

Figure 12.27: Particles Represented by Spheres



The following collision force laws are available:

- Spring
- Spring-dashpot
- Hertzian
- Hertzian-dashpot
- Friction
- Rolling friction

The size of the spring constant of the normal contact force for a given collision pair should at least satisfy the condition that for the biggest parcels in that collision pair and the highest relative velocity, the spring constant should be high enough to make two parcels in a collision recoil with a maximum overlap that is not too large compared to the parcel diameter. You can estimate the value of the spring constant k using the following equation:

$$k=\frac{\pi v^2}{3\varepsilon_D^2} D\rho \tag{12.491}$$

where D is the parcel diameter, ρ is the particle mass density, v is the relative velocity between two colliding particles, and ε_D is the fraction of the diameter for allowable overlap. The collision time scale is evaluated as $\pi\sqrt{\frac{m}{k}}$, where m is the parcel mass (defined by $\frac{\rho D^3 \pi}{6}$)

12.15.1.1. The Spring Collision Law

For the linear spring collision law, a unit vector ($\vec{e}_{12}$) is defined from particle 1 to particle 2:

$$\vec{e}_{12}= \left(x_2-x_1\right) \Big/ \|x_2-x_1\| \tag{12.492}$$

where x_1 and x_2 represent the position of particle 1 and 2, respectively.

The overlap δ (which is less than zero during contact) is defined as follows:

$$\delta= \|x_2-x_1\|- \left(r_1+ r_2\right) \tag{12.493}$$

where r_1 and r_2 represent the radii of particle 1 and 2, respectively.

The force on particle 1 ($\vec{F}_1$) is then calculated using a spring constant K that you define (and that must be greater than zero):

$$\vec{F}_1= K\delta\vec{e}_{12} \tag{12.494}$$

and then by Newton's third law, the force on particle 2 ($\vec{F}_2$) is:

$$\vec{F}_2= -\vec{F}_1 \tag{12.495}$$

Note that $\vec{F}_1$ is directed away from particle 2, because δ is less than zero for contact.

12.15.1.2. The Spring-Dashpot Collision Law

The spring-dashpot collision law is a linear spring force law as described previously, augmented with a dashpot term described below.

For the spring-dashpot collision law, you define a spring constant K as in the spring collision law, along with a coefficient of restitution for the dashpot term (η). Note that $0<\eta\leq 1$.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

In preparation for the force calculations, the following expressions are evaluated:

$$f_{\text{loss}}=\sqrt{\pi^2+\ln^2\eta} \tag{12.496}$$

$$m_{12}=\frac{m_1 m_2}{m_1+m_2} \tag{12.497}$$

$$t_{\text{coll}}=f_{\text{loss}}\sqrt{\frac{m_{12}}{K}} \tag{12.498}$$

$$\gamma=-2\frac{m_{12}\ln\eta}{t_{\text{coll}}} \tag{12.499}$$

$$\vec{v}_{12}=\vec{v}_2-\vec{v}_1 \tag{12.500}$$

where f_{loss} is a loss factor, m_1 and m_2 are the masses of particle 1 and 2, respectively, m_{12} is the so-called "reduced mass", t_{coll} is the collision time scale, $\vec{v}_1$ and $\vec{v}_2$ are the velocities of particle 1 and 2, respectively, $\vec{v}_{12}$ is the relative velocity, and γ is the damping coefficient. Note that $\gamma \geq 0$, because $\ln\eta \leq 0$.

With the previous expressions, the force on particle 1 can be calculated as:

$$\vec{F}_1=(K\delta+\gamma(\vec{v}_{12}\cdot\vec{e}_{12}))\vec{e}_{12} \tag{12.501}$$

$\overrightarrow{F_2}$ is calculated using Equation 12.495 (p. 578).

12.15.1.3. The Hertzian Collision Law

The Hertzian collision law [239] (p. 1066) is a nonlinear collision law. Using the same notations as in the section The Spring Collision Law (p. 578), the force on particle 1 can be described as:

$$\vec{F}_1= K_H\delta^{3/2}\vec{e}_{12} \tag{12.502}$$

Here, the constant K_H is calculated from the respective Young's Moduli E_1 and E_2 of the two colliding particles and their Poisson's ratios ν_1 and ν_2:

$$K_H=\frac{4}{3}\frac{E_1E_2}{E_2\left(1-\nu_1^2\right)+E_1\left(1-\nu_2^2\right)}\sqrt{\frac{r_1r_2}{r_1+r_2}} \tag{12.503}$$

The Young's Modulus has units of Pascals and is normally in the range of 1 GPa to a few 100 GPa. The Poisson ratio is a dimensionless constant in the range of -1 to 0.5.

$\overrightarrow{F_2}$ is calculated using Equation 12.495 (p. 578).

12.15.1.4. The Hertzian-Dashpot Collision Law

The Hertzian-dashpot collision law is a nonlinear collision force law as described in the section The Hertzian Collision Law (p. 579) augmented with the same dashpot term as in the linear spring-dashpot collision law (see The Spring-Dashpot Collision Law (p. 578)). That is, Equation 12.502 (p. 579) is modified as follows:

$$\vec{F}_1= \left(K_H\delta^{3/2}+\gamma\left(\vec{v}_{12}\cdot\vec{e}_{12}\right)\right)\vec{e}_{12} \tag{12.504}$$

$\overrightarrow{F_2}$ is calculated using Equation 12.495 (p. 578).

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

12.15.1.5. The Friction Collision Law

The friction collision law is based on the equation for Coulomb friction (F_{friction}):

$$F_{\text{friction}}=\mu F_{\text{normal}} \tag{12.505}$$

where μ is the friction coefficient and F_{normal} is the magnitude of the normal to the surface force. The direction of the friction force is opposite to the relative tangential motion, and may or may not inhibit the relative tangential motion depending on the following:

- the size of the tangential momentum
- the size of other tangential forces (for example, tangential components from gravity and drag)

The friction coefficient is a function of the relative tangential velocity magnitude (v_{rt}):

- $v_r \le v_{\text{glide}}$:
  - $\mu(v_r)=\mu_{\text{stick}}+\left(\mu_{\text{stick}}-\mu_{\text{glide}}\right)\left(v_r/v_{\text{glide}}-2.0\right)\left(v_r/v_{\text{glide}}\right)$
- $v_{\text{glide}} < v_r \le v_{\text{limit}}$:
  - $\mu(v_r)=\mu_{\text{glide}}$
- $v_r > v_{\text{limit}}$:
  - $v_{\text{ratio}}=\dfrac{\left(v_r-v_{\text{limit}}\right)}{slope_{\text{limit}}}$
  - $\mu_{\text{ratio}}=\dfrac{\mu_{\text{glide}}}{\mu_{\text{limit}}}$
  - $\mu(v_r)=\mu_{\text{glide}}\dfrac{1+v_{\text{ratio}}}{1+\mu_{\text{ratio}}v_{\text{ratio}}}$

where

μ_{stick} is the sticking friction coefficient

μ_{glide} is the gliding friction coefficient

μ_{limit} is the high velocity limit friction coefficient

v_{glide} is the gliding velocity — for lower velocities, μ is interpolated quadratically between μ_{stick} and μ_{glide}

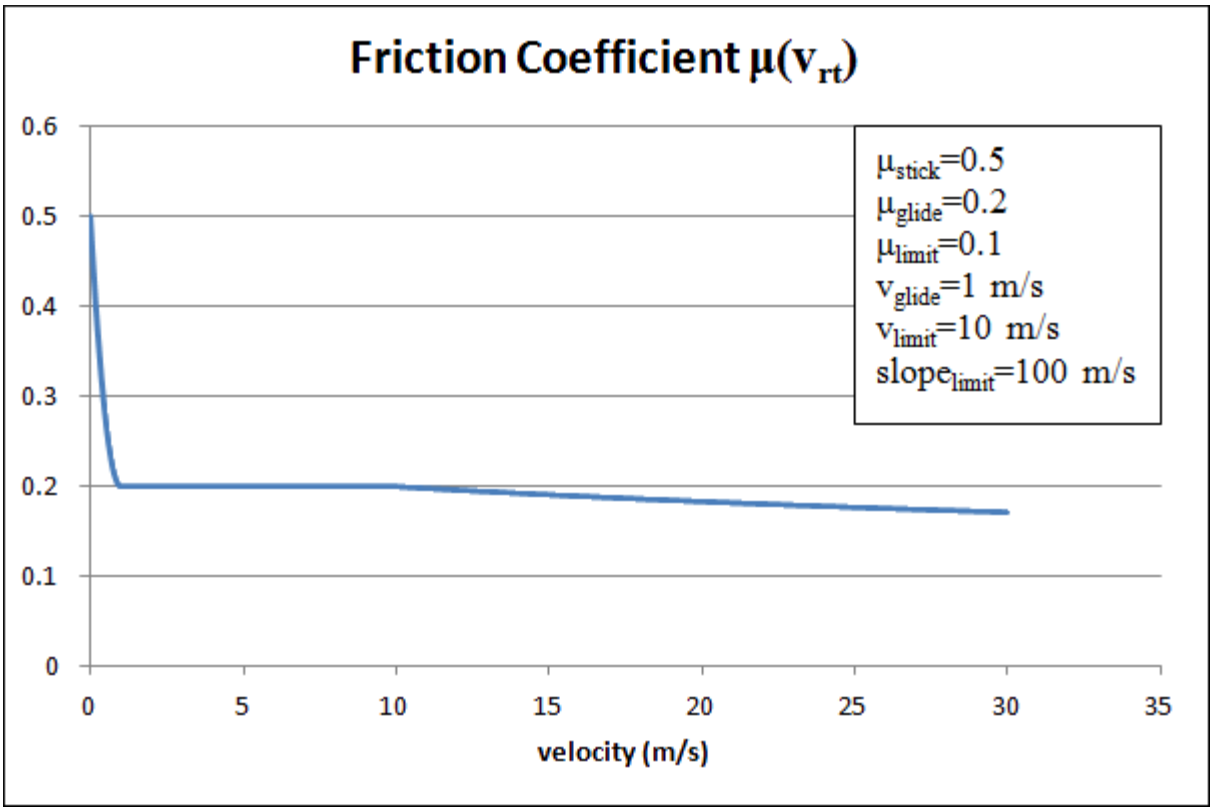
v_{limit} is the limit velocity — for higher velocities, $\mu(v_r)$ approaches μ_{limit}

$slope_{\text{limit}}$ is a parameter determining the how fast $\mu(v_r)$ approaches μ_{limit}

For an example of a plot of $\mu(v_{\text{rt}})$, see Figure 12.28: An Example of a Friction Coefficient Plot (p. 581).

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Figure 12.28: An Example of a Friction Coefficient Plot



12.15.1.6. Rolling Friction Collision Law for DEM

The rolling friction collision law is an extension of the friction collision law (The Friction Collision Law (p. 580)) based on the equation for Coulomb friction ($F_{rolling}$):

$$F_{rolling}=\mu_{rolling}F_{normal} \tag{12.506}$$

where $\mu_{rolling}$ is the rolling friction coefficient, and F_{normal} is the magnitude of the force that is either normal to the particle surface or pointing from one particle center to another. The rolling friction force $\vec{F}_{rolling}$ acts only on the local torque at the particle-particle or particle-wall contact point. This force may or may not inhibit the relative angular velocity, depending on the size of the relative torque.

12.15.1.7. DEM Parcels

For typical applications, the computational cost of tracking all of the particles individually is prohibitive. Instead, the approach for the discrete element method is similar to that of the DPM, in that like particles are divided into parcels, and then the position of each parcel is determined by tracking a single representative particle. The DEM approach differs from the DPM in the following ways:

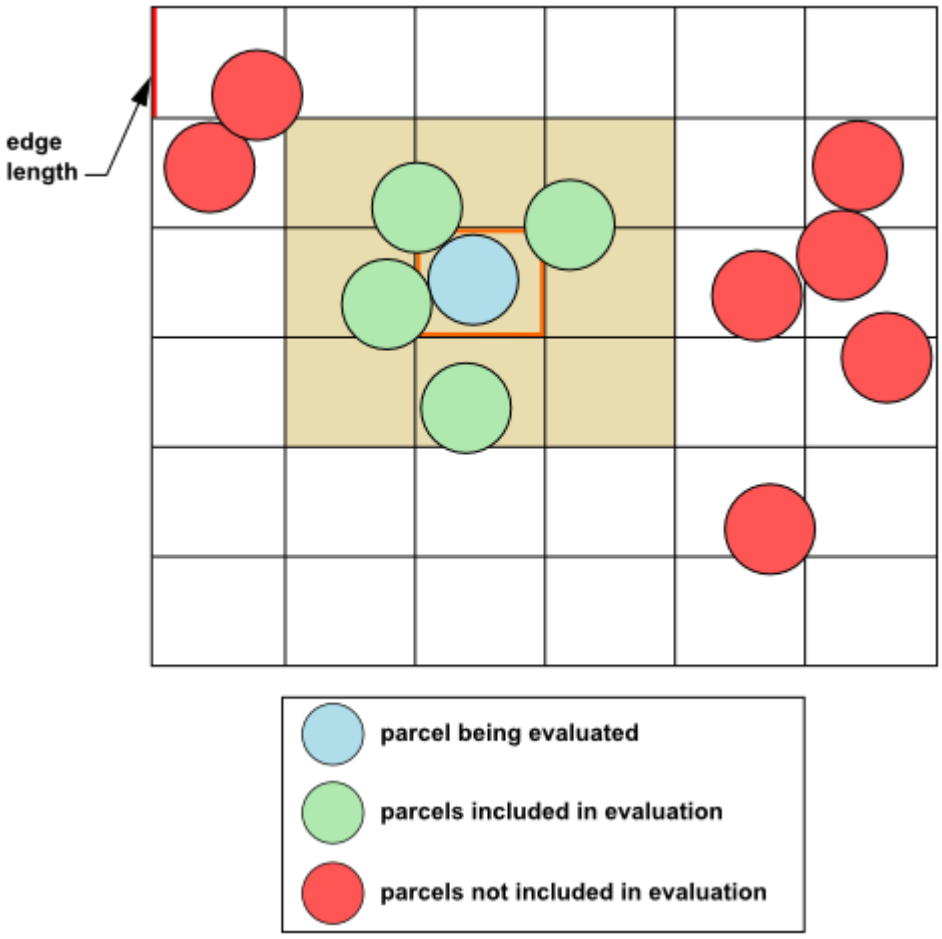
- The mass used in the DEM calculations of the collisions is that of the entire parcel, not just that of the single representative particle.
- The radius of the DEM parcel is that of a sphere whose volume is the mass of the entire parcel divided by the particle density.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

12.15.1.8. Cartesian Collision Mesh

When evaluating the collisions between parcels, it is too costly to conduct a direct force evaluation that involves all of the parcels. Consider that for N parcels, the number of pairs that would need to be inspected for every time step would be on the order of N^2. To address this issue, a geometric approach is used: the domain is divided by a suitable Cartesian mesh (where the edge length of the mesh cells is a multiple of the largest parcel diameter), and then the force evaluation is only conducted for parcels that are in neighboring mesh cells, because particles in more remote cells of the collision mesh are *a priori* known to be out of reach. See Figure 12.29: Force Evaluation for Parcels (p. 582) for an illustration.

Figure 12.29: Force Evaluation for Parcels



12.16. One-Way and Two-Way Coupling

You can use Ansys Fluent to predict the discrete phase patterns based on a fixed continuous phase flow field (an *uncoupled* approach or "one-way coupling"), or you can include the effect of the discrete phase on the continuum (a *coupled* approach or "two-way coupling"). In the coupled approach, the continuous phase flow pattern is impacted by the discrete phase (and vice versa), and you can alternate calculations of the continuous phase and discrete phase equations until a converged coupled solution is achieved.

For more information, see the following sections:

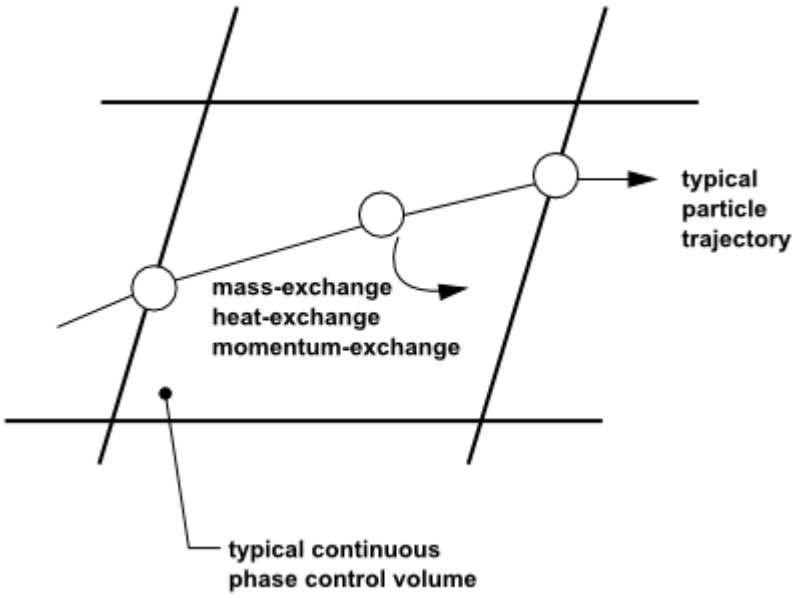
Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

12.16.1. Coupling Between the Discrete and Continuous Phases
12.16.2. Momentum Exchange
12.16.3. Heat Exchange
12.16.4. Mass Exchange
12.16.5. Under-Relaxation of the Interphase Exchange Terms
12.16.6. Interphase Exchange During Stochastic Tracking

12.16.1. Coupling Between the Discrete and Continuous Phases

As the trajectory of a particle is computed, Ansys Fluent keeps track of the heat, mass, and momentum gained or lost by the particle stream that follows that trajectory and these quantities can be incorporated in the subsequent continuous phase calculations. Thus, while the continuous phase always impacts the discrete phase, you can also incorporate the effect of the discrete phase trajectories on the continuum. This two-way coupling is accomplished by alternately solving the discrete and continuous phase equations until the solutions in both phases have stopped changing. This interphase exchange of heat, mass, and momentum from the particle to the continuous phase is depicted qualitatively in Figure 12.30: Heat, Mass, and Momentum Transfer Between the Discrete and Continuous Phases (p. 583). Note that no interchange terms are computed for particles defined as massless, where the discrete phase trajectories have no impact on the continuum.

Figure 12.30: Heat, Mass, and Momentum Transfer Between the Discrete and Continuous Phases



12.16.2. Momentum Exchange

The momentum transfer from the continuous phase to the discrete phase is computed in Ansys Fluent by examining the change in momentum of a particle as it passes through each control volume in the Ansys Fluent model. This momentum change is computed as

$$F = \sum \left(\frac{18 \mu C_D Re}{\rho_p d_p^2 24} \left(u_p - u \right) + F_{other} \right) \dot{m}_p \Delta t \tag{12.507}$$

where

μ = viscosity of the fluid

ρ_p = density of the particle

d_p = diameter of the particle

Re = relative Reynolds number

u_p = velocity of the particle

u = velocity of the fluid

C_D = drag coefficient

$\dot{m}_p$ = mass flow rate of the particles

Δt = time step

F_{other} = other interaction forces

This momentum exchange appears as a momentum source in the continuous phase momentum balance in any subsequent calculations of the continuous phase flow field and can be reported by Ansys Fluent as described in Postprocessing for the Discrete Phase in the User's Guide.

12.16.3. Heat Exchange

The heat transfer from the discrete phase to the continuous phase is computed in Ansys Fluent by examining the change in thermal energy of a particle as it passes through each control volume in the Ansys Fluent model. In the absence of a chemical reaction (that is, for all particle laws except Law 5) the heat exchange is computed as

$$Q = \frac{\dot{m}_{p,0}}{m_{p,0}} \left(\left(m_{p_{in}} - m_{p_{out}} \right) \left(-H_{lat_{ref}} + H_{pyrol} \right) + m_{p_{in}} \int_{T_{ref}}^{T_{p_{in}}} c_{p_p} dT - m_{p_{out}} \int_{T_{ref}}^{T_{p_{out}}} c_{p_p} dT \right) \tag{12.508}$$

where

$\dot{m}_{p,0}$ = initial mass flow rate of the particle injection (kg/s)

$m_{p,0}$ = initial mass of the particle (kg)

$m_{p_{in}}$ = mass of the particle on cell entry (kg)

$m_{p_{out}}$ = mass of the particle on cell exit (kg)

c_{p_p} = heat capacity of the particle (J/kg-K)

H_{pyrol} = heat of pyrolysis as volatiles are evolved (J/kg)

$T_{p_{in}}$ = temperature of the particle on cell entry (K)

$T_{p_{out}}$ = temperature of the particle on cell exit (K)

T_{ref} = reference temperature for enthalpy (K)

$H_{lat_{ref}}$ = latent heat at reference conditions (J/kg)

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

The latent heat at the reference conditions $H_{lat_{ref}}$ for droplet particles is computed as the difference of the liquid and gas standard formation enthalpies, and can be related to the latent heat at the boiling point as follows:

$$H_{lat_{ref}} = H_{lat} - \int_{T_{ref}}^{T_{bp}} c_{p_g} dT + \int_{T_{ref}}^{T_{bp}} c_{p_p} dT \tag{12.509}$$

where

c_{p_g} = heat capacity of gas product species (J/kg-K)

T_{bp} = boiling point temperature (K)

H_{lat} = latent heat at the boiling point temperature (J/kg)

For the volatile part of the combusting particles, some constraints are applied to ensure that the enthalpy source terms do not depend on the particle history. The formulation should be consistent with the mixing of two gas streams, one consisting of the fluid and the other consisting of the volatiles. Hence $H_{lat_{ref}}$ is derived by applying a correction to H_{lat}, which accounts for different heat capacities in the particle and gaseous phase:

$$H_{lat_{ref}} = H_{lat} - \int_{T_{p,init}}^{T_{ref}} c_{p_g} dT + \int_{T_{p,init}}^{T_{ref}} c_{p_p} dT \tag{12.510}$$

where

$T_{p,init}$ = particle initial temperature (K)

12.16.4. Mass Exchange

The mass transfer from the discrete phase to the continuous phase is computed in Ansys Fluent by examining the change in mass of a particle as it passes through each control volume in the Ansys Fluent model. The mass change is computed simply as

$$M = \frac{\Delta m_p}{m_{p,0}} \dot{m}_{p,0} \tag{12.511}$$

This mass exchange appears as a source of mass in the continuous phase continuity equation and as a source of a chemical species defined by you. The mass rates per cell (in kg/s) are included in any subsequent calculations of the continuous phase flow field and are reported by Ansys Fluent as described in Postprocessing for the Discrete Phase in the User's Guide.

12.16.5. Under-Relaxation of the Interphase Exchange Terms

The interphase exchange of momentum, heat, and mass is under-relaxed during the calculation, so that:

$$F_{new} = F_{old} + \alpha \left(F_{calculated} - F_{old} \right) \tag{12.512}$$

$$Q_{new} = Q_{old} + \alpha \left(Q_{calculated} - Q_{old} \right) \tag{12.513}$$

$$M_{new} = M_{old} + \alpha \left(M_{calculated} - M_{old} \right) \tag{12.514}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

where α is the under-relaxation factor for particles/droplets.

For additional information about under-relaxation of the interphase exchange terms, see Under-Relaxation of the Interphase Exchange Terms in the *Fluent User's Guide*.

12.16.6. Interphase Exchange During Stochastic Tracking

When stochastic tracking is performed, the interphase exchange terms, computed via Equation 12.507 (p. 583) to Equation 12.514 (p. 585), are computed for each stochastic trajectory with the particle mass flow rate, $\dot{m}_{p0}$, divided by the number of stochastic tracks computed. This implies that an equal mass flow of particles follows each stochastic trajectory.

12.17. Node Based Averaging

Various DPM parcel quantities can affect the continuous fluid flow by being integrated into the flow solver equations. By default, Ansys Fluent applies the effects of a DPM parcel only to the fluid cell that contains the parcel. As an alternative, mesh node averaging can be used to distribute the parcel's effects to neighboring mesh nodes. This reduces the grid dependency of DPM simulations, since the parcel's effects on the flow solver are distributed more smoothly across neighboring cells.

A variable φ is averaged on a mesh node using the following equation:

$$\overline{\varphi}_{node} = \sum_{k} N_{in\,parcel} w\left(\vec{x}_p^{\,k} - \vec{x}_{node}\right) \varphi_p \tag{12.515}$$

Here φ_p is the particle variable, $N_{in\,parcel}$ is the number in the parcel, and $\overline{\varphi}_{node}$ is the accumulation of the particle variable on the node for all parcels k. The function w is a weighting function, or "kernel."

The following kernels are available in Ansys Fluent:

- **nodes-per-cell**

$$w_k\left(\vec{x}_p^{\,k} - \vec{x}_{node}\right) = \frac{1}{n_{nodes}} \tag{12.516}$$

- **shortest-distance**

$$w_k\left(\vec{x}_p^{\,k} - \vec{x}_{node}\right) = \frac{1 - \frac{\left|\vec{x}_p^{\,k} - \vec{x}_{node}\right|}{\sum_i \left|\vec{x}_p^{\,i} - \vec{x}_{node}\right|}}{\sum_j \left(1 - \frac{\left|\vec{x}_p^{\,j} - \vec{x}_{node}\right|}{\sum_i \left|\vec{x}_p^{\,i} - \vec{x}_{node}\right|}\right)} \tag{12.517}$$

- **inverse-distance**

$$w_k\left(\vec{x}_p^{\,k} - \vec{x}_{node}\right) = \frac{\frac{1}{\left|\vec{x}_p^{\,k} - \vec{x}_{node}\right|}}{\sum_i \left(\frac{1}{\left|\vec{x}_p^{\,i} - \vec{x}_{node}\right|}\right)} \tag{12.518}$$

- **gaussian**

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$w\left(\vec{x}_p^k - \vec{x}_{node}\right) = \left(\frac{a}{\pi}\right)^{3/2} \exp\left(-a\frac{\left|\vec{x}_p^k - \vec{x}_{node}\right|^2}{\Delta x^2}\right) \tag{12.519}$$

where Δx is a characteristic length scale of the cell containing the parcel.

For the Gaussian kernel, an additional parameter a is necessary to control the width of the Gaussian distribution. Kaufmann et al. [293] (p. 1070) had good success with a small width and using a=6, while Apte et al. [25] (p. 1054) used a broad distribution and a value of a=0.5. The default value is a=1.

For information on using node based averaging, refer to Node Based Averaging of Particle Data in the User's Guide.

12.18. Blockage Effect

In standard Lagrangian-Eulerian predictions, particles are assumed to be mass points and, therefore, do not displace the carrier phase. The volume fraction of the carrier phase is assumed to be unity throughout the entire domain, regardless of the actual particle concentration. Physically, the assumption of particles sharing the same physical space with the carrier phases only holds true if the volume fraction of the Lagrangian phase is negligible. In many technically relevant simulations, the volume fraction of the local particle phase may not be small, and the blocking effect of the particulate phase on the carrier phase may need to be taken into account.

The blocking effect impact requires the flow around the particles to be incorporated into transport equations. Any transported variable φ belonging to a carrier phase is represented by:

$$\frac{\delta}{\delta t}\left(\rho\varphi\right) + \nabla\left(\rho U\varphi\right) = \nabla\left(\rho\Gamma\nabla\varphi\right) + S \tag{12.520}$$

where

ρ = carrier phase density

Γ = kinematic combined (laminar + turbulent) diffusivity

U = velocity of the carrier phase

S = optional source per unit volume

The blocking effect changes the transport equation to:

$$\frac{\delta}{\delta t}\left(\rho\alpha_f\varphi\right) + \nabla\left(\rho\alpha_f U\varphi\right) = \nabla.\left(\alpha_f\rho\Gamma\nabla\varphi\right) + \alpha_f S \tag{12.521}$$

where α_f is the carrier phase volume fraction.

The carrier phase volume fraction α_f is computed from the particle phase volume fraction α_p as:

$$\alpha_f = 1 - \min\left(\alpha_p, \alpha_{p,max}\right) \tag{12.522}$$

where $\alpha_{p,max}$ is a user-specified value. This value is clipped between 0 and 1.

The DPM blocking model modifies the single-phase equations to incorporate the effect of blocking due to the appearance of the carrier phase volume fraction in the transient, convection, diffusion, and source

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

terms of the corresponding transport equation. Note that particle source terms in the carrier phase are still computed using the "mass-point" assumption, that is, no scaling of the particle source terms is performed based on the volume fraction of the carrier phase.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Chapter 13: Modeling Macroscopic Particles

The Macroscopic Particle Model (MPM) is a UDF-based quasi-direct numerical approach for tracking macroscopic particles [488] (p. 1081). The MPM is applicable to Lagrangian particulate flows that cannot be solved using conventional point-mass particle models. In such flows, the particle size cannot be neglected. In these situations, particle volume must be considered when modeling hydrodynamics and wall effects.

The MPM model provides a special treatment that accounts for the following:

- Flow blockage and momentum exchange
- Calculation of drag and torque on particles
- Particle-particle and particle-wall collision, and friction dynamics
- Particle deposition and buildup
- Particle-particle and particle-wall attraction forces

In the MPM approach [8] (p. 1053), each macroscopic particle spans several computational cells. Each particle is represented by a sphere with six degrees of freedom to account for the particle translational and rotational motion. The particle is injected in the flow domain at the beginning of a flow time step. The particle is assumed to be touching a computational cell during the time step if one or more nodes of the cell are located inside the particle volume. Each particle transport equation is solved in a Lagrangian reference frame.

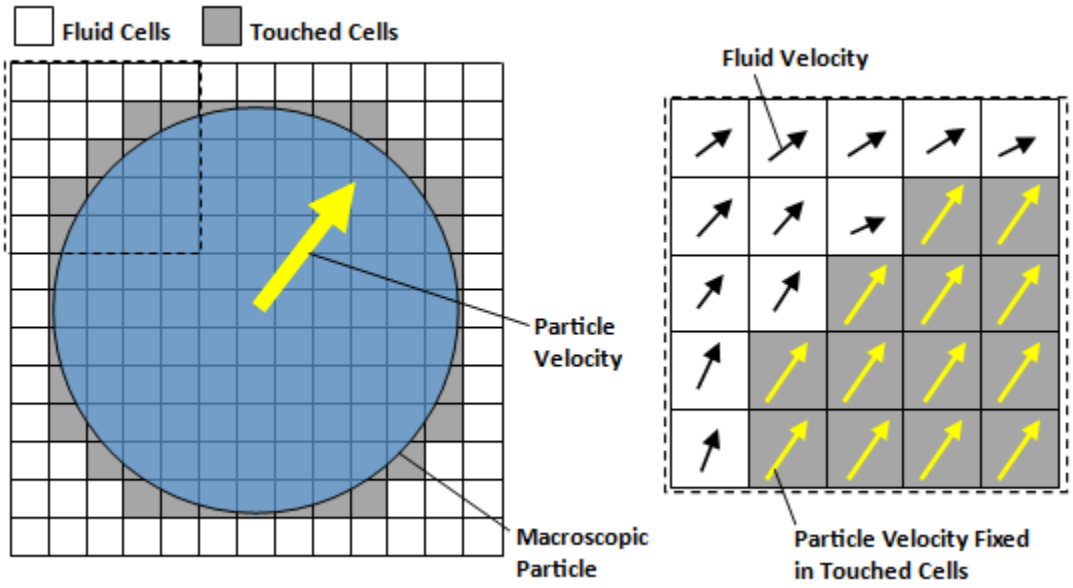
13.1. Momentum Transfer to Fluid Flow
13.2. Fluid Forces and Torques on Particle
13.3. Particle/Particle and Particle/Wall Collisions
13.4. Field Forces
13.5. Particle Deposition and Buildup

13.1. Momentum Transfer to Fluid Flow

At each time step, a volume-fraction weighted velocity between the particle velocity (translational and rotational) and the flow velocity in the cell from the last time step is assigned to the fluid cells touched by the particle. Additionally, the cell velocity of all cells touched by the particle is forced towards particle velocity extrapolated to the cell center using source terms. As a consequence, the flow velocity of touched cells is dominated by the particle velocity

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Figure 13.1: Fixing Velocities in Fluid Cells Touched by the Particle



13.2. Fluid Forces and Torques on Particle

At each time step, drag and torque acting on the particle are computed using explicit expressions involving the velocity, pressure, and shear stress distribution in the fluid cells surrounding the particle [9] (p. 1053). The total fluid forces and torques acting on a macroscopic particle in the direction **i** consist of virtual mass, pressure, and viscous fluid components:

$$R_i = R_{m,i} + R_{p,i} + R_{v,i} \tag{13.1}$$

The i^{th} virtual mass component of the fluid force and torque experienced by a particle, $R_{m,i}$, is calculated as the integral of the rate of change of momentum for all fluid cells within a particle volume:

$$R_{m,i} = \left(\sum_{\substack{volume \\ cells}} m_f V_{f,i} - \sum_{\substack{volume \\ cells}} m_f V_{p,i} \right) \frac{1}{\Delta t} \tag{13.2}$$

where m_f is the cell fluid mass; $V_{f,i}$ and $V_{p,i}$ are the fluid and particle velocities in the direction i, respectively; and Δt is the flow time step.

The i^{th} pressure component of the fluid force and torque acting on the particle surface, $R_{p,i}$, is calculated based on the pressure distribution around the particle:

$$R_{p,i} = \sum_{\substack{surface \\ cells}} P d^2 \frac{r_i}{|\vec{r}|} \tag{13.3}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

where P is the pressure, d^2 is the approximated area of a particle surface in a fluid cell touching the particle, $\vec{r}$ is the radius vector from the fluid cell center to the particle center, and r_i is the Cartesian component of vector $\vec{r}$ in the i^{th} direction.

The i^{th} viscous component of the fluid force and torque acting on a particle surface, $R_{v,i}$, is calculated based on the shear stress distribution around the particle:

$$R_{v,i} = \sum_{\substack{surface \\ cells}} \sum_{j} \vec{\tau}_{ij} d^2 \left(-\frac{r_j}{|\vec{r}|} \right) \tag{13.4}$$

where $\vec{\tau_{ji}}$ is the shear stress in the positive i^{th} direction acting on a plane perpendicular to the j^{th} direction, and r_j is the Cartesian component of vector $\vec{r}$ in the j^{th} direction.

Based on fluid forces and torques, the new particle position, velocities, and accelerations are calculated at each flow time step.

13.3. Particle/Particle and Particle/Wall Collisions

Ansys Fluent provides a hard sphere collision algorithm (billiard ball model) to account for particle-particle and particle-wall collisions. All collisions are assumed to be binary and quasi-instantaneous, and with contact occurring at a single point. The algorithm considers impulse forces and momentum experienced by particles during collision and it also accounts for energy dissipation.

The motions of two particles at the time of the collision are expressed as:

$$\begin{aligned} m_1\left(V_1 - V_1^0\right) &= J \\ m_2\left(V_2 - V_2^0\right) &= -J \\ \frac{I_1}{r_1}\left(\omega_1 - \omega_1^0\right) &= J \times n \\ \frac{I_2}{r_2}\left(\omega_2 - \omega_2^0\right) &= J \times n \end{aligned} \tag{13.5}$$

where,

1 and 2 subscripts refer to particles participating in the collision
m is the particle mass
I is the particle moment of inertia
V and ω are the linear and angular particle velocities, respectively
0 superscript refers to particle velocities before the collision
r is the particle radius
J is the impulse force
n is the unit vector in the normal direction

The impulse force in the normal direction is expressed by:

$$J_n = -\left(1 + e_n\right) \frac{V_{12}^0 \cdot n}{B} \tag{13.6}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

The impulse force in the tangential direction for sticking collision is expressed as:

$$J_{t,sticking} = -\left(1+e_t\right)\frac{V_{12}^0 \cdot t}{B} \tag{13.7}$$

The impulse force in the tangential direction for sliding collision is expressed as:

$$J_{t,sliding} = -\mu_f J_n \tag{13.8}$$

In the above equations:

e_n and e_t are the normal and tangential coefficients of restitution, respectively

μ_f is the friction coefficient

t is the unit vector in the tangential direction

$B = 1/m_1 + 1/m_2$

V_{12}^0 is the relative surface velocity (rotational and translational) of the two particles

13.4. Field Forces

The gravity and Buoyancy forces are automatically accounted for in the Ansys Fluent MPM model.

Other particle-particle field forces, such as electrostatic, magnetic or cohesive forces, are implemented in the MPM using a potential force model in which inter-particle field forces acting on the particle are defined as:

$$F_i = \sum_j \frac{G_p M_i^{n1} M_j^{n2}}{R_{i-j}^{n3}} \tag{13.9}$$

where,

M_i and M_j = masses of the interacting particles i and j

R_{i-j} = distance between the interacting particles i and j

G_p, n_1, n_2, and n_3 = user-specified particle constants

The model constants G_p, n_1, n_2, and n_3 can be indirectly related to the tensile strength or cohesive forces in solids.

In a similar manner, particle-wall field forces are defined as:

$$F_{i,wall} = \sum_{walls} \frac{G_w M_i^{n4}}{R_i^{n5}} \tag{13.10}$$

where,

R_i = the closest distance of the particle from the wall w

G_w, n_4, n_5, = user-specified particle constants

The signs of the constants G_p and G_w determine whether the particle-particle and particle-wall field forces are attraction or repulsion forces.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

13.5. Particle Deposition and Buildup

For filtration/separation-type applications, phenomena of particle deposition and buildup on selected surfaces are implemented based on the critical impact velocity algorithm. If the particle velocity is below the user-specified critical impact velocity, the particle will adhere to the wall upon impact. If the particle velocity is above the critical impact velocity, the particle will rebound off the wall. The deposited particles are assigned zero velocities and accelerations.

The deposition model also allows for detachment of the particle if the fluid force exceeds a critical user-specified limit.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information
of ANSYS, Inc. and its subsidiaries and affiliates.

Chapter 14: Multiphase Flows

This chapter discusses the general multiphase models that are available in Ansys Fluent. Introduction (p. 595) provides a brief introduction to multiphase modeling, Discrete Phase (p. 447) discusses the Lagrangian dispersed phase model, and Solidification and Melting (p. 817) describes Ansys Fluent's model for solidification and melting. For information about using the general multiphase models in Ansys Fluent, see Modeling Multiphase Flows in the User's Guide. Information about the various theories behind the multiphase models is presented in the following sections:

14.1. Introduction

14.2. Choosing a General Multiphase Model

14.3. Volume of Fluid (VOF) Model Theory

14.4. Mixture Model Theory

14.5. Eulerian Model Theory

14.6. Wet Steam Model Theory

14.7. Modeling Mass Transfer in Multiphase Flows

14.8. Modeling Species Transport in Multiphase Flows

14.1. Introduction

A large number of flows encountered in nature and technology are a mixture of phases. Physical phases of matter are gas, liquid, and solid, but the concept of phase in a multiphase flow system is applied in a broader sense. In multiphase flow, a phase can be defined as an identifiable class of material that has a particular inertial response to and interaction with the flow and the potential field in which it is immersed. For example, different-sized solid particles of the same material can be treated as different phases because each collection of particles with the same size will have a similar dynamical response to the flow field.

Information is organized into the following subsections:

14.1.1. Multiphase Flow Regimes

14.1.2. Examples of Multiphase Systems

14.1.1. Multiphase Flow Regimes

Multiphase flow regimes can be grouped into four categories: gas-liquid or liquid-liquid flows; gas-solid flows; liquid-solid flows; and three-phase flows.

14.1.1.1. Gas-Liquid or Liquid-Liquid Flows

The following regimes are gas-liquid or liquid-liquid flows:

- Bubbly flow: This is the flow of discrete gaseous or fluid bubbles in a continuous fluid.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

- Droplet flow: This is the flow of discrete fluid droplets in a continuous gas.
- Slug flow: This is the flow of large bubbles in a continuous fluid.
- Stratified/free-surface flow: This is the flow of immiscible fluids separated by a clearly-defined interface.

See Figure 14.1: Multiphase Flow Regimes (p. 597) for illustrations of these regimes.

14.1.1.2. Gas-Solid Flows

The following regimes are gas-solid flows:

- Particle-laden flow: This is flow of discrete particles in a continuous gas.
- Pneumatic transport: This is a flow pattern that depends on factors such as solid loading, Reynolds numbers, and particle properties. Typical patterns are dune flow, slug flow, and homogeneous flow.
- Fluidized bed: This consists of a vessel containing particles, into which a gas is introduced through a distributor. The gas rising through the bed suspends the particles. Depending on the gas flow rate, bubbles appear and rise through the bed, intensifying the mixing within the bed.

See Figure 14.1: Multiphase Flow Regimes (p. 597) for illustrations of these regimes.

14.1.1.3. Liquid-Solid Flows

The following regimes are liquid-solid flows:

- Slurry flow: This flow is the transport of particles in liquids. The fundamental behavior of liquid-solid flows varies with the properties of the solid particles relative to those of the liquid. In slurry flows, the Stokes number (see Equation 14.4 (p. 602)) is normally less than 1. When the Stokes number is larger than 1, the characteristic of the flow is liquid-solid fluidization.
- Hydrotransport: This describes densely-distributed solid particles in a continuous liquid.
- Sedimentation: This describes a tall column initially containing a uniform dispersed mixture of particles. At the bottom, the particles will slow down and form a sludge layer. At the top, a clear interface will appear, and in the middle a constant settling zone will exist.

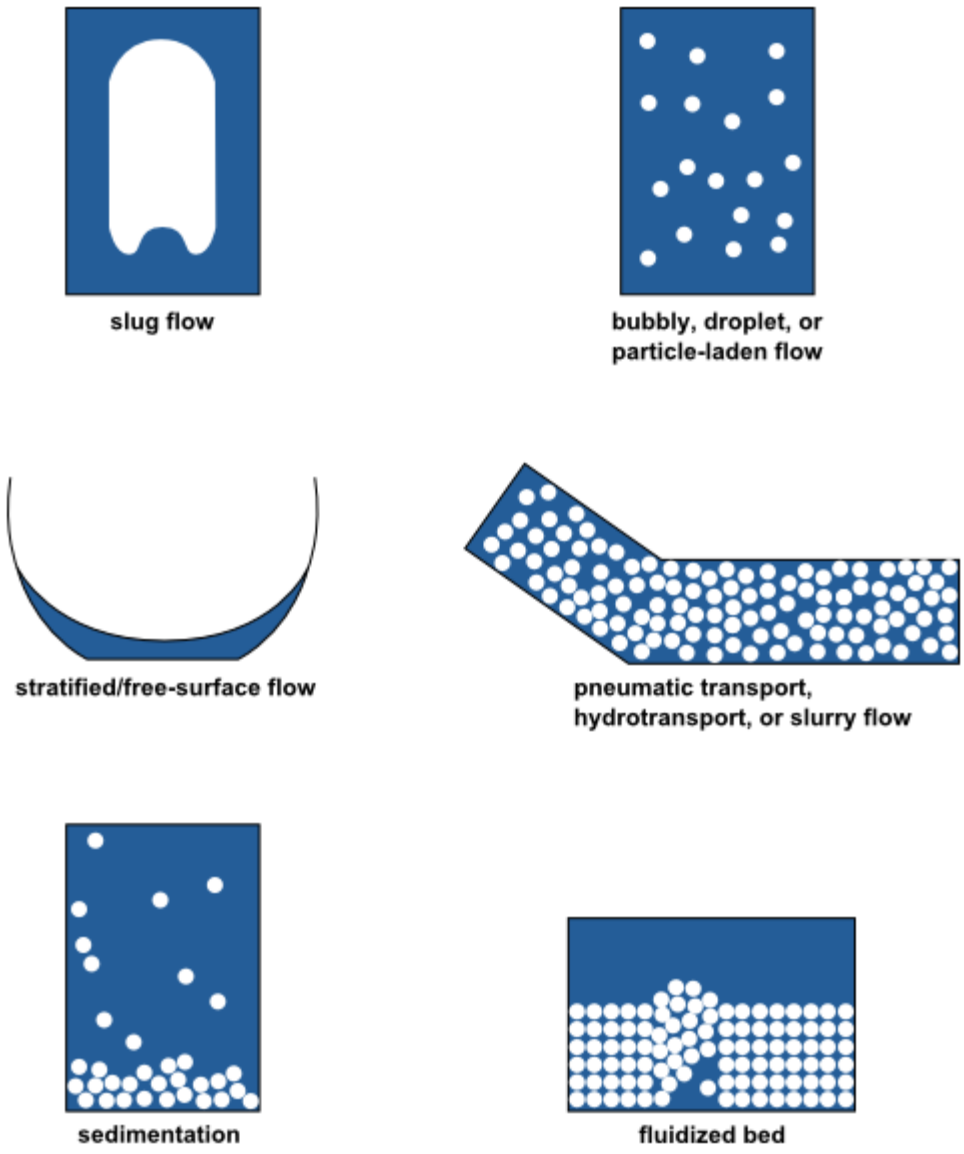
See Figure 14.1: Multiphase Flow Regimes (p. 597) for illustrations of these regimes.

14.1.1.4. Three-Phase Flows

Three-phase flows are combinations of the other flow regimes listed in the previous sections.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Figure 14.1: Multiphase Flow Regimes



14.1.2. Examples of Multiphase Systems

Specific examples of each regime described in Multiphase Flow Regimes (p. 595) are listed below:

- Bubbly flow examples include absorbers, aeration, air lift pumps, cavitation, evaporators, flotation, and scrubbers.
- Droplet flow examples include absorbers, atomizers, combustors, cryogenic pumping, dryers, evaporation, gas cooling, and scrubbers.
- Slug flow examples include large bubble motion in pipes or tanks.
- Stratified/free-surface flow examples include sloshing in offshore separator devices and boiling and condensation in nuclear reactors.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information
of ANSYS, Inc. and its subsidiaries and affiliates.

- Particle-laden flow examples include cyclone separators, air classifiers, dust collectors, and dust-laden environmental flows.
- Pneumatic transport examples include transport of cement, grains, and metal powders.
- Fluidized bed examples include fluidized bed reactors and circulating fluidized beds.
- Slurry flow examples include slurry transport and mineral processing
- Hydrotransport examples include mineral processing and biomedical and physiochemical fluid systems
- Sedimentation examples include mineral processing.

14.2. Choosing a General Multiphase Model

The first step in solving any multiphase problem is to determine which of the regimes described in Multiphase Flow Regimes (p. 595) best represents your flow. Model Comparisons (p. 599) provides some broad guidelines for determining appropriate models for each regime, and Detailed Guidelines (p. 600) provides details about how to determine the degree of interphase coupling for flows involving bubbles, droplets, or particles, and the appropriate model for different amounts of coupling.

Information is organized into the following subsections:

14.2.1. Approaches to Multiphase Modeling
14.2.2. Model Comparisons
14.2.3. Time Schemes in Multiphase Flow
14.2.4. Stability and Convergence

14.2.1. Approaches to Multiphase Modeling

Advances in computational fluid mechanics have provided the basis for further insight into the dynamics of multiphase flows. Currently there are two approaches for the numerical calculation of multiphase flows: the Euler-Lagrange approach (discussed in Introduction (p. 447)) and the Euler-Euler approach (discussed in the following section).

14.2.1.1. The Euler-Euler Approach

In the Euler-Euler approach, the different phases are treated mathematically as interpenetrating continua. Since the volume of a phase cannot be occupied by the other phases, the concept of phasic volume fraction is introduced. These volume fractions are assumed to be continuous functions of space and time and their sum is equal to one. Conservation equations for each phase are derived to obtain a set of equations, which have similar structure for all phases. These equations are closed by providing constitutive relations that are obtained from empirical information, or, in the case of granular flows , by application of kinetic theory.

In Ansys Fluent, three different Euler-Euler multiphase models are available: the volume of fluid (VOF) model, the mixture model, and the Eulerian model.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

14.2.1.1.1. The VOF Model

The VOF model (described in Volume of Fluid (VOF) Model Theory (p. 604)) is a surface-tracking technique applied to a fixed Eulerian mesh. It is designed for two or more immiscible fluids where the position of the interface between the fluids is of interest. In the VOF model, a single set of momentum equations is shared by the fluids, and the volume fraction of each of the fluids in each computational cell is tracked throughout the domain. Applications of the VOF model include stratified flows, free-surface flows, filling, sloshing, the motion of large bubbles in a liquid, the motion of liquid after a dam break, the prediction of jet breakup (surface tension), and the steady or transient tracking of any liquid-gas interface.

14.2.1.1.2. The Mixture Model

The mixture model (described in Mixture Model Theory (p. 640)) is designed for two or more phases (fluid or particulate). As in the Eulerian model, the phases are treated as interpenetrating continua. The mixture model solves for the mixture momentum equation and prescribes relative velocities to describe the dispersed phases. Applications of the mixture model include particle-laden flows with low loading, bubbly flows, sedimentation , and cyclone separators. The mixture model can also be used without relative velocities for the dispersed phases to model homogeneous multiphase flow.

14.2.1.1.3. The Eulerian Model

The Eulerian model (described in Eulerian Model Theory (p. 659)) is the most complex of the multiphase models in Ansys Fluent. It solves a set of n momentum and continuity equations for each phase. Coupling is achieved through the pressure and interphase exchange coefficients. The manner in which this coupling is handled depends upon the type of phases involved; granular (fluid-solid) flows are handled differently than nongranular (fluid-fluid) flows. For granular flows, the properties are obtained from application of kinetic theory. Momentum exchange between the phases is also dependent upon the type of mixture being modeled. Ansys Fluent's user-defined functions allow you to customize the calculation of the momentum exchange. Applications of the Eulerian multiphase model include bubble columns, risers, particle suspension, and fluidized beds.

14.2.2. Model Comparisons

In general, once you have determined the flow regime that best represents your multiphase system, you can select the appropriate model based on the following guidelines:

- For bubbly, droplet, and particle-laden flows in which the phases mix and/or dispersed-phase volume fractions exceed 10%, use either the mixture model (described in Mixture Model Theory (p. 640)) or the Eulerian model (described in Eulerian Model Theory (p. 659)).
- For slug flows, use the VOF model. See Volume of Fluid (VOF) Model Theory (p. 604) for more information about the VOF model.
- For stratified/free-surface flows, use the VOF model. See Volume of Fluid (VOF) Model Theory (p. 604) for more information about the VOF model.
- For pneumatic transport, use the mixture model for homogeneous flow (described in Mixture Model Theory (p. 640)) or the Eulerian model for granular flow (described in Eulerian Model Theory (p. 659)).

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

- For fluidized beds, use the Eulerian model for granular flow. See Eulerian Model Theory (p. 659) for more information about the Eulerian model.
- For slurry flows and hydrotransport, use the mixture or Eulerian model (described, respectively, in Mixture Model Theory (p. 640), Eulerian Model Theory (p. 659)).
- For sedimentation, use the Eulerian model. See Eulerian Model Theory (p. 659) for more information about the Eulerian model.
- For general, complex multiphase flows that involve multiple flow regimes, select the aspect of the flow that is of most interest, and choose the model that is most appropriate for that aspect of the flow. Note that the accuracy of results will not be as good as for flows that involve just one flow regime, since the model you use will be valid for only part of the flow you are modeling.

As discussed in this section, the VOF model is appropriate for stratified or free-surface flows, and the mixture and Eulerian models are appropriate for flows in which the phases mix or separate and/or dispersed-phase volume fractions exceed 10%. (Flows in which the dispersed-phase volume fractions are less than or equal to 10% can be modeled using the discrete phase model described in Discrete Phase (p. 447).)

To choose between the mixture model and the Eulerian model, you should consider the following guidelines:

- If there is a wide distribution of the dispersed phases (that is, if the particles vary in size and the largest particles do not separate from the primary flow field), the mixture model may be preferable (that is, less computationally expensive). If the dispersed phases are concentrated just in portions of the domain, you should use the Eulerian model instead.
- If interphase drag laws that are applicable to your system are available (either within Ansys Fluent or through a user-defined function), the Eulerian model can usually provide more accurate results than the mixture model. Even though you can apply the same drag laws to the mixture model, as you can for a non-granular Eulerian simulation, if the interphase drag laws are unknown or their applicability to your system is questionable, the mixture model may be a better choice. For most cases with spherical particles, the Schiller-Naumann law is more than adequate. For cases with non-spherical particles, a user-defined function can be used.
- If you want to solve a simpler problem, which requires less computational effort, the mixture model may be a better option, since it solves a smaller number of equations than the Eulerian model. If accuracy is more important than computational effort, the Eulerian model is a better choice. However, the complexity of the Eulerian model can make it less computationally stable than the mixture model.

Ansys Fluent's multiphase models are compatible with Ansys Fluent's dynamic mesh modeling feature. For more information on the dynamic mesh feature, see Flows Using Sliding and Dynamic Meshes (p. 35). For more information about how other Ansys Fluent models are compatible with Ansys Fluent's multiphase models, see Appendix A: Ansys Fluent Model Compatibility in the User's Guide.

14.2.2.1. Detailed Guidelines

For stratified and slug flows, the choice of the VOF model, as indicated in Model Comparisons (p. 599), is straightforward. Choosing a model for the other types of flows is less straightforward. As a general guide, there are some parameters that help to identify the appropriate multiphase model for

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

these other flows: the particulate loading, β, and the Stokes number, St. (Note that the word "particle" is used in this discussion to refer to a particle, droplet, or bubble.)

14.2.2.1.1. The Effect of Particulate Loading

Particulate loading has a major impact on phase interactions. The particulate loading is defined as the mass density ratio of the dispersed phase (d) to that of the carrier phase (c):

$$\beta = \frac{\alpha_d \rho_d}{\alpha_c \rho_c} \tag{14.1}$$

The material density ratio

$$\gamma = \frac{\rho_d}{\rho_c} \tag{14.2}$$

is greater than 1000 for gas-solid flows, about 1 for liquid-solid flows, and less than 0.001 for gas-liquid flows.

Using these parameters it is possible to estimate the average distance between the individual particles of the particulate phase. An estimate of this distance has been given by Crowe et al. [124] (p. 1060):

$$\frac{L}{d_d} = \left(\frac{\pi}{6} \frac{1+\kappa}{\kappa} \right)^{1/3} \tag{14.3}$$

where $\kappa = \frac{\beta}{\gamma}$. Information about these parameters is important for determining how the dispersed phase should be treated. For example, for a gas-particle flow with a particulate loading of 1, the interparticle space $\frac{L}{d_d}$ is about 8; the particle can therefore be treated as isolated (that is, very low particulate loading).

Depending on the particulate loading, the degree of interaction between the phases can be divided into the following three categories:

- For very low loading, the coupling between the phases is one-way (that is, the fluid carrier influences the particles via drag and turbulence, but the particles have no influence on the fluid carrier). The discrete phase (Discrete Phase (p. 447)), mixture, and Eulerian models can all handle this type of problem correctly. Since the Eulerian model is the most expensive, the discrete phase or mixture model is recommended.

- For intermediate loading, the coupling is two-way (that is, the fluid carrier influences the particulate phase via drag and turbulence, but the particles in turn influence the carrier fluid via reduction in mean momentum and turbulence). The discrete phase (Discrete Phase (p. 447)), mixture, and Eulerian models are all applicable in this case, but you need to take into account other factors in order to decide which model is more appropriate. See below for information about using the Stokes number as a guide.

- For high loading, there is two-way coupling plus particle pressure and viscous stresses due to particles (four-way coupling). Only the Eulerian model will handle this type of problem correctly.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

14.2.2.1.2. The Significance of the Stokes Number

For systems with intermediate particulate loading, estimating the value of the Stokes number can help you select the most appropriate model. The Stokes number can be defined as the relation between the particle response time and the system response time:

$$St = \frac{\tau_d}{t_s} \tag{14.4}$$

where $\tau_d = \frac{\rho_d d_d^2}{18\mu_c}$ and t_s is based on the characteristic length (L_s) and the characteristic velocity (V_s) of the system under investigation: $t_s = \frac{L_s}{V_s}$.

For $St \ll 1.0$, the particle will follow the flow closely and any of the three models (discrete phase(Discrete Phase (p. 447)), mixture, or Eulerian) is applicable; you can therefore choose the least expensive (the mixture model, in most cases), or the most appropriate considering other factors. For $St > 1.0$, the particles will move independently of the flow and either the discrete phase model (Discrete Phase (p. 447)) or the Eulerian model is applicable. For $St \approx 1.0$, again any of the three models is applicable; you can choose the least expensive or the most appropriate considering other factors.

14.2.2.1.2.1. Examples

For a coal classifier with a characteristic length of 1 m and a characteristic velocity of 10 m/s, the Stokes number is 0.04 for particles with a diameter of 30 microns, but 4.0 for particles with a diameter of 300 microns. Clearly the mixture model will not be applicable to the latter case.

For the case of mineral processing, in a system with a characteristic length of 0.2 m and a characteristic velocity of 2 m/s, the Stokes number is 0.005 for particles with a diameter of 300 microns. In this case, you can choose between the mixture and Eulerian models. The volume fractions are too high for the discrete phase model (Discrete Phase (p. 447)), as noted below.

14.2.2.1.3. Other Considerations

The use of the discrete phase model (Discrete Phase (p. 447)) is limited to low volume fractions, unless you are using the dense discrete phase model formulation. In addition, for the discrete phase model simulation, you can choose many more advanced combustion models compared to the Eulerian models. To account for particle distributions, you will need to use the population balance models (see the Population Balance Module Manual) or the discrete phase model and the dense discrete phase model.

14.2.3. Time Schemes in Multiphase Flow

In many multiphase applications, the process can vary spatially as well as temporally. In order to accurately model multiphase flow, both higher-order spatial and time discretization schemes are necessary. In addition to the first-order time scheme in Ansys Fluent, the second-order time scheme is available in the Mixture and Eulerian multiphase models, and with the VOF Implicit Scheme.

Important:

The second-order time scheme cannot be used with the VOF Explicit Schemes.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

The second-order time scheme has been adapted to all the transport equations, including mixture phase momentum equations, energy equations, species transport equations, turbulence models, phase volume fraction equations, the pressure correction equation, and the granular flow model. In multiphase flow, a general transport equation (similar to that of Equation 24.16 (p. 969)) may be written as

$$\frac{\partial(\alpha\rho\varphi)}{\partial t}+\nabla\cdot\left(\alpha\rho\vec{V}\varphi\right)=\nabla\cdot\overline{\overline{\tau}}+S_{\varphi} \tag{14.5}$$

Where φ is either a mixture (for the mixture model) or a phase variable, α is the phase volume fraction (unity for the mixture equation), ρ is the mixture phase density, $\vec{V}$ is the mixture or phase velocity (depending on the equations), $\overline{\overline{\tau}}$ is the diffusion term, and S_{φ} is the source term.

As a fully implicit scheme, this second-order time-accurate scheme achieves its accuracy by using an Euler backward approximation in time (see Equation 24.24 (p. 971)). The general transport equation, Equation 14.5 (p. 603) is discretized as

$$\frac{3\left(\alpha_p\rho_p\varphi_p Vol\right)^{n+1}-4\left(\alpha_p\rho_p\varphi_p Vol\right)^{n}+\left(\alpha_p\rho_p\varphi_p\right)^{n-1}}{2\Delta t}=\sum\left[A_{nb}\left(\varphi_{nb}-\varphi_p\right)\right]^{n+1}+S_U{}^{n+1}-S_p{}^{n+1}\varphi_p{}^{n+1} \tag{14.6}$$

Equation 14.6 (p. 603) can be written in simpler form:

$$A_p\varphi_p=\sum A_n b\varphi_n b+S_{\varphi} \tag{14.7}$$

where

$$A_p=\sum A_{nb}{}^{n+1}+S_p{}^{n+1}+\frac{1.5\left(\alpha_p\rho_p Vol\right)^{n+1}}{\Delta t}$$

$$S_{\varphi}=S_U{}^{n+1}+\frac{2\left(\alpha_p\rho_p\varphi_p Vol\right)^{n}-0.5\left(\alpha_p\rho_p\varphi_p Vol\right)^{n-1}}{\Delta t}$$

This scheme is easily implemented based on Ansys Fluent's existing first-order Euler scheme. It is unconditionally stable, however, the negative coefficient at the time level t_{n-1}, of the three-time level method, may produce oscillatory solutions if the time steps are large.

This problem can be eliminated if a bounded second-order scheme is introduced. See Bounded Second-Order Implicit Time Integration (p. 969) for more information.

14.2.4. Stability and Convergence

The process of solving a multiphase system is inherently difficult and you may encounter some stability or convergence problems.

For a steady solution: It is recommended that you use the Multiphase Coupled solver, described in detail in Coupled Solution for Eulerian Multiphase Flows in the *Fluent User's Guide*. The iterative nature of this solver requires a good starting patched field. If difficulties are encountered due to higher order schemes, or due to the complexities of the problem, you may need to reduce the Courant number. The default Courant number is 200 but it can be reduced to as low as 4. This can later be increased if the iteration process runs smoothly. In addition, there are explicit under-relaxation factors for velocities and pressure. All other under-relaxation factors are implicit. Lower under-relaxation factors for the volume fraction equation may delay the solution dramatically with the Coupled solver (any value

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

0.5 or above is adequate); on the contrary, PC SIMPLE would normally need a low under-relaxation for the volume fraction equation.

For a time-dependent solution: A proper initial field is required to avoid instabilities, which usually arise from poor initial fields. If the CPU time is a concern for transient problems, then the best option is to use PC SIMPLE. When body forces are significant, or if the solution requires higher order numerical schemes, it is recommended that you start with a small time step, which can be increased after performing a few time steps to get a better approximation of the pressure field.

When computing unsteady flows using Non-Iterative Time Advancement (NITA), good initial conditions are important. Stability problems may arise in models with poor meshes or in the presence of large body forces.

If you are using the MRF model for steady-state or quasi-steady analysis and are experiencing convergence problems, you can switch to the unsteady solver and attempt to converge to a steady solution. When using NITA with the MRF model, you should be aware of NITA robustness problems due to poor mesh quality or large source terms in the momentum equations at the MRF boundaries. The Iterative Time Advancement (ITA) is preferable for MRF simulations as it gives you more control over the number of iterations per time step.

In addition, Ansys Fluent offers a Full Multiphase Coupled solver where all velocities, pressure correction and volume fraction correction are solved simultaneously, which currently is not as robust as the others.

Furthermore, Ansys Fluent has an option to solve stratified immiscible fluids within the Eulerian multiphase formulation. This feature is similar to the single fluid VOF solution, but in the context of multiple velocities.

14.3. Volume of Fluid (VOF) Model Theory

Information is organized into the following subsections:

14.3.1. Overview of the VOF Model
14.3.2. Limitations of the VOF Model
14.3.3. Steady-State and Transient VOF Calculations
14.3.4. Volume Fraction Equation
14.3.5. Material Properties
14.3.6. Momentum Equation
14.3.7. Energy Equation
14.3.8. Additional Scalar Equations
14.3.9. Surface Tension and Adhesion
14.3.10. Open Channel Flow
14.3.11. Open Channel Wave Boundary Conditions
14.3.12. Coupled Level-Set and VOF Model

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

14.3.1. Overview of the VOF Model

The VOF model can model two or more immiscible fluids by solving a single set of momentum equations and tracking the volume fraction of each of the fluids throughout the domain. Typical applications include the prediction of jet breakup, the motion of large bubbles in a liquid, the motion of liquid after a dam break, and the steady or transient tracking of any liquid-gas interface.

14.3.2. Limitations of the VOF Model

The following restrictions apply to the VOF model in Ansys Fluent:

- You must use the pressure-based solver. The VOF model is not available with the density-based solver.
- All control volumes must be filled with either a single fluid phase or a combination of phases. The VOF model does not allow for void regions where no fluid of any type is present.
- Streamwise periodic flow with specified mass flow rate cannot be modeled when the VOF model is used.
- The second-order implicit time-stepping formulation cannot be used with the VOF explicit scheme.
- When tracking particles with the DPM model in combination with the VOF model, the **Shared Memory** method cannot be selected (Parallel Processing for the Discrete Phase Model). (Note that using the **Message Passing** or **Hybrid** method enables the compatibility of all multiphase flow models with the DPM model.)
- The VOF model is not compatible with non-premixed, partially premixed, and premixed combustion models.

14.3.3. Steady-State and Transient VOF Calculations

The VOF formulation in Ansys Fluent is generally used to compute a time-dependent solution, but for problems in which you are concerned only with a steady-state solution, it is possible to perform a steady-state calculation. A steady-state VOF calculation is sensible only when your solution is independent of the initial conditions and there are distinct inflow boundaries for the individual phases. For example, since the shape of the free surface inside a rotating cup depends on the initial level of the fluid, such a problem must be solved using the time-dependent formulation. On the other hand, the flow of water in a channel with a region of air on top and a separate air inlet can be solved with the steady-state formulation.

The VOF formulation relies on the fact that two or more fluids (or phases) are not interpenetrating. For each additional phase that you add to your model, a variable is introduced: the volume fraction of the phase in the computational cell. In each control volume, the volume fractions of all phases sum to unity. The fields for all variables and properties are shared by the phases and represent volume-averaged values, as long as the volume fraction of each of the phases is known at each location. Thus the variables and properties in any given cell are either purely representative of one of the phases, or representative of a mixture of the phases, depending upon the volume fraction values. In other words, if the q^{th} fluid's volume fraction in the cell is denoted as α_q, then the following three conditions are possible:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

- $\alpha_q = 0$: The cell is empty (of the q^{th} fluid).
- $\alpha_q = 1$: The cell is full (of the q^{th} fluid).
- $0 < \alpha_q < 1$: The cell contains the interface between the q^{th} fluid and one or more other fluids.

Based on the local value of α_q, the appropriate properties and variables will be assigned to each control volume within the domain.

14.3.4. Volume Fraction Equation

The tracking of the interface(s) between the phases is accomplished by the solution of a continuity equation for the volume fraction of one (or more) of the phases. For the q^{th} phase, this equation has the following form:

$$\frac{1}{\rho_q}\left[\frac{\partial}{\partial t}\left(\alpha_q \rho_q\right) + \nabla \cdot \left(\alpha_q \rho_q \vec{v}_q\right) = S_{\alpha_q} + \sum_{p=1}^{n}\left(\dot{m}_{pq} - \dot{m}_{qp}\right)\right] \tag{14.8}$$

where $\dot{m}_{qp}$ is the mass transfer from phase q to phase p and $\dot{m}_{pq}$ is the mass transfer from phase p to phase q. By default, the source term on the right-hand side of Equation 14.8 (p. 606), S_{α_q}, is zero, but you can specify a constant or user-defined mass source for each phase. See Modeling Mass Transfer in Multiphase Flows (p. 751) for more information on the modeling of mass transfer in Ansys Fluent's general multiphase models.

The volume fraction equation will not be solved for the primary phase; the primary-phase volume fraction will be computed based on the following constraint:

$$\sum_{q=1}^{n} \alpha_q = 1 \tag{14.9}$$

The volume fraction equation may be solved either through implicit or explicit time formulation.

14.3.4.1. The Implicit Formulation

When the implicit formulation is used, the volume fraction equation is discretized in the following manner:

$$\frac{\alpha_q^{n+1}\rho_q^{n+1} - \alpha_q^n \rho_q^n}{\Delta t} V + \sum_f \left(\rho_q^{n+1} U_f^{n+1} \alpha_{q,f}^{n+1}\right) = \left[S_{\alpha_q} + \sum_{p=1}^{n}\left(\dot{m}_{pq} - \dot{m}_{qp}\right)\right] V \tag{14.10}$$

where:

$n+1$ = index for current time step

n = index for previous time step

α_q^{n+1} = cell value of volume fraction at time step n+1

α_q^n = cell value of volume fraction at time step n

$\alpha_{q,f}^{n+1}$ = face value of the q^{th} volume fraction at time step n+1

U_f^{n+1} = volume flux through the face at time step n+1

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

V = cell volume

Since the volume fraction at the current time step is a function of other quantities at the current time step, a scalar transport equation is solved iteratively for each of the secondary-phase volume fractions at each time step.

Face fluxes are interpolated using the chosen spatial discretization scheme. The schemes available in Ansys Fluent for the implicit formulation are discussed in Spatial Discretization Schemes for Volume Fraction in the User's Guide.

The implicit formulation can be used for both time-dependent and steady-state calculations. See Choosing Volume Fraction Formulation in the User's Guide for details.

14.3.4.2. The Explicit Formulation

The explicit formulation is time-dependent and the volume fraction is discretized in the following manner:

$$\frac{\alpha_q^{n+1}\rho_q^{n+1}-\alpha_q^n\rho_q^n}{\Delta t}V+\sum_f\left(\rho_q U_f^n \alpha_{q,f}^n\right)=\left[\sum_{p=1}^{n}\left(\dot{m}_{pq}-\dot{m}_{qp}\right)+S_{\alpha_q}\right]V \tag{14.11}$$

where

$n+1$ = index for new (current) time step

n = index for previous time step

$\alpha_{q,f}$ = face value of the q^{th} volume fraction

V = volume of cell

U_f = volume flux through the face, based on normal velocity

Since the volume fraction at the current time step is directly calculated based on known quantities at the previous time step, the explicit formulation does not require and iterative solution of the transport equation during each time step.

The face fluxes can be interpolated using interface tracking or capturing schemes such as Geo-Reconstruct, CICSAM, Compressive, and Modified HRIC (See Interpolation Near the Interface (p. 608)). The schemes available in Ansys Fluent for the explicit formulation are discussed in Spatial Discretization Schemes for Volume Fraction in the *Fluent User's Guide*.

Ansys Fluent automatically refines the time step for the integration of the volume fraction equation, but you can influence this time step calculation by modifying the Courant number. You can choose to update the volume fraction once for each time step, or once for each iteration within each time step. These options are discussed in more detail in Setting Time-Dependent Parameters for the Explicit Volume Fraction Formulation in the *Fluent User's Guide*.

Important:

When the explicit scheme is used, a time-dependent solution must be computed.

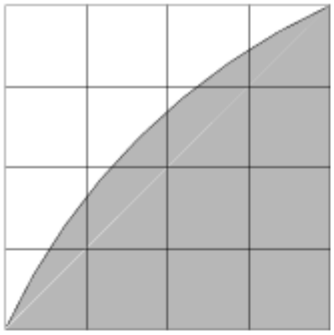
Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

14.3.4.3. Interpolation Near the Interface

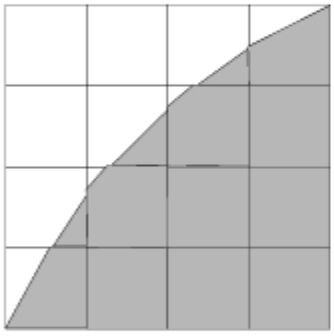
Ansys Fluent's control-volume formulation requires that convection and diffusion fluxes through the control volume faces be computed and balanced with source terms within the control volume itself.

In the geometric reconstruction and donor-acceptor schemes, Ansys Fluent applies a special interpolation treatment to the cells that lie near the interface between two phases. Figure 14.2: Interface Calculations (p. 608) shows an actual interface shape along with the interfaces assumed during computation by these two methods.

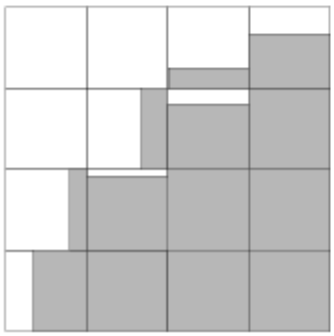
Figure 14.2: Interface Calculations



actual interface shape



interface shape represented by the geometric reconstruction (piecewise-linear) scheme



interface shape represented by the donor-acceptor scheme

The explicit scheme and the implicit scheme treat these cells with the same interpolation as the cells that are completely filled with one phase or the other (that is, using the standard upwind (First-Order Upwind Scheme (p. 963)), second-order (Second-Order Upwind Scheme (p. 963)), QUICK (QUICK Scheme (p. 966)), modified HRIC (Modified HRIC Scheme (p. 967)), compressive (The Compressive Scheme and Interface-Model-based Variants (p. 610)), or CICSAM scheme (The Compressive Inter-

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

face Capturing Scheme for Arbitrary Meshes (CICSAM) (p. 610)), rather than applying a special treatment.

14.3.4.3.1. The Geometric Reconstruction Scheme

In the geometric reconstruction approach, the standard interpolation schemes that are used in Ansys Fluent are used to obtain the face fluxes whenever a cell is completely filled with one phase or another. When the cell is near the interface between two phases, the geometric reconstruction scheme is used.

The geometric reconstruction scheme represents the interface between fluids using a piecewise-linear approach. In Ansys Fluent this scheme is the most accurate and is applicable for general unstructured meshes. The geometric reconstruction scheme is generalized for unstructured meshes from the work of Youngs [725] (p. 1095). It assumes that the interface between two fluids has a linear slope within each cell, and uses this linear shape for calculation of the advection of fluid through the cell faces. (See Figure 14.2: Interface Calculations (p. 608).)

The first step in this reconstruction scheme is calculating the position of the linear interface relative to the center of each partially-filled cell, based on information about the volume fraction and its derivatives in the cell. The second step is calculating the advecting amount of fluid through each face using the computed linear interface representation and information about the normal and tangential velocity distribution on the face. The third step is calculating the volume fraction in each cell using the balance of fluxes calculated during the previous step.

Important:

When the geometric reconstruction scheme is used, a time-dependent solution must be computed. Also, if you are using a conformal mesh (that is, if the mesh node locations are identical at the boundaries where two subdomains meet), you must ensure that there are no two-sided (zero-thickness) walls within the domain. If there are, you will need to slit them, as described in Slitting Face Zones in the User's Guide.

14.3.4.3.2. The Donor-Acceptor Scheme

In the donor-acceptor approach, the standard interpolation schemes that are used in Ansys Fluent are used to obtain the face fluxes whenever a cell is completely filled with one phase or another. When the cell is near the interface between two phases, a "donor-acceptor" scheme is used to determine the amount of fluid advected through the face [248] (p. 1067). This scheme identifies one cell as a donor of an amount of fluid from one phase and another (neighbor) cell as the acceptor of that same amount of fluid, and is used to prevent numerical diffusion at the interface. The amount of fluid from one phase that can be convected across a cell boundary is limited by the minimum of two values: the filled volume in the donor cell or the free volume in the acceptor cell.

The orientation of the interface is also used in determining the face fluxes. The interface orientation is either horizontal or vertical, depending on the direction of the volume fraction gradient of the q^{th} phase within the cell, and that of the neighbor cell that shares the face in question. Depending

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

on the interface's orientation as well as its motion, flux values are obtained by pure upwinding, pure downwinding, or some combination of the two.

Important:

When the donor-acceptor scheme is used, a time-dependent solution must be computed. Also, the donor-acceptor scheme can be used only with quadrilateral or hexahedral meshes. In addition, if you are using a conformal mesh (that is, if the mesh node locations are identical at the boundaries where two subdomains meet), you must ensure that there are no two-sided (zero-thickness) walls within the domain. If there are, you will need to slit them, as described in Slitting Face Zones in the User's Guide.

14.3.4.3.3. The Compressive Interface Capturing Scheme for Arbitrary Meshes (CICSAM)

The compressive interface capturing scheme for arbitrary meshes (CICSAM), based on Ubbink's work [659] (p. 1091), is a high resolution differencing scheme. The CICSAM scheme is particularly suitable for flows with high ratios of viscosities between the phases. CICSAM is implemented in Ansys Fluent as an explicit scheme and offers the advantage of producing an interface that is almost as sharp as the geometric reconstruction scheme.

14.3.4.3.4. The Compressive Scheme and Interface-Model-based Variants

The compressive scheme is a second order reconstruction scheme based on the slope limiter. The slope limiters are used in spatial discretization schemes to avoid the spurious oscillations or wiggles that would otherwise occur with high order spatial discretization schemes due to sharp changes in the solution domain. The theory below is applicable to the zonal discretization, the phase localized discretization and the regime based discretization (available with the flow regime modeling), which use the framework of the compressive scheme.

$$\alpha_f = \alpha_d + \beta \nabla \alpha_d \cdot d\vec{r} \tag{14.12}$$

where

α_f = face VOF value

α_d = donor cell VOF value

β = slope limiter value

$\nabla \alpha_d$ = donor cell VOF gradient value

$d\vec{r}$ = cell to face distance

The slope limiter is constrained to values between 0 and 2 (inclusive). For values less than 1, the spatial discretization is represented by a low resolution scheme. For values between 1 and 2, the spatial discretization is represented by a high resolution scheme. The slope limiter values and their discretization schemes are shown in the table below.

Table 14.1: Slope Limiter Values and Their Discretization Schemes

Slope Limiter Value β	Scheme
0	first order upwind

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Slope Limiter Value β	Scheme
1	second order reconstruction bounded by the global minimum/maximum of the volume fraction
2	compressive
$0<\beta<1$ and $1<\beta<2$	blended: where a value between 0 and 1 means blending of the first order and second order and a value between 1 and 2 means blending of the second order and compressive scheme

The compressive scheme discretization depends on the selection of interface regime type. When sharp interface regime modeling is selected, the compressive scheme is only suited to modeling sharp interfaces. However, when sharp/dispersed interface modeling is chosen, the compressive scheme is appropriate for both sharp and dispersed interface modeling.

14.3.4.3.5. Bounded Gradient Maximization (BGM)

The BGM scheme is introduced to obtain sharp interfaces with the VOF model, comparable to that obtained by the Geometric Reconstruction scheme. Currently this scheme is available only with the steady-state solver and cannot be used for transient problems. In the BGM scheme, discretization occurs in such a way so as to maximize the local value of the gradient, by maximizing the degree to which the face value is weighted towards the extrapolated downwind value [683] (p. 1092).

14.3.5. Material Properties

The properties appearing in the transport equations are determined by the presence of the component phases in each control volume. In a two-phase system, for example, if the phases are represented by the subscripts 1 and 2, and if the volume fraction of the second of these is being tracked, the density in each cell is given by

$$\rho=\alpha_2\rho_2+\left(1-\alpha_2\right)\rho_1 \tag{14.13}$$

In general, for an n-phase system, the volume-fraction-averaged density takes on the following form:

$$\rho=\sum\alpha_q\rho_q \tag{14.14}$$

All other properties (for example, viscosity) are computed in this manner.

14.3.6. Momentum Equation

A single momentum equation is solved throughout the domain, and the resulting velocity field is shared among the phases. The momentum equation, shown below, is dependent on the volume fractions of all phases through the properties ρ and μ.

$$\frac{\partial}{\partial t}\left(\rho\vec{v}\right)+\nabla\cdot\left(\rho\vec{v}\vec{v}\right)=-\nabla p+\nabla\cdot\left[\mu\left(\nabla\vec{v}+\nabla\vec{v}^T\right)\right]+\rho\vec{g}+\vec{F} \tag{14.15}$$

One limitation of the shared-fields approximation is that in cases where large velocity differences exist between the phases, the accuracy of the velocities computed near the interface can be adversely affected.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Note that if the viscosity ratio is more than $1x10^3$, this may lead to convergence difficulties. The compressive interface capturing scheme for arbitrary meshes (CICSAM) (The Compressive Interface Capturing Scheme for Arbitrary Meshes (CICSAM) (p. 610)) is suitable for flows with high ratios of viscosities between the phases, therefore solving the problem of poor convergence.

14.3.7. Energy Equation

The energy equation, also shared among the phases, is shown below.

$$\frac{\partial}{\partial t}(\rho E)+\nabla\cdot\left(\vec{v}(\rho E+p)\right)=\nabla\cdot\left(k_{eff}\nabla T-\sum_q\sum_j h_{j,q}\vec{J}_{j,q}+\left(\bar{\bar{\tau}}_{eff}\cdot\vec{v}\right)\right)+S_h \tag{14.16}$$

where k_{eff} is the effective conductivity ($k+k_t$, where k_t is the turbulent thermal conductivity, defined according to the turbulence model being used), $\vec{J}_j$ is the diffusion flux of species j, $h_{j,q}$ is enthalpy of species j in phase q, and $\vec{J}_{j,q}$ is diffusive flux of species j in phase q. The first three terms on the right-hand side of Equation 14.16 (p. 612) represent energy transfer due to conduction, species diffusion, and viscous dissipation, respectively. S_h includes volumetric heat sources that you have defined but not the heat sources generated by finite-rate volumetric or surface reactions since species formation enthalpy is already included in the total enthalpy calculation as described in Energy Sources Due to Reaction (p. 174).

The VOF model treats energy, E, as a mass-averaged variable:

$$E=\frac{\sum_{q=1}^{n}\alpha_q\rho_q E_q}{\sum_{q=1}^{n}\alpha_q\rho_q} \tag{14.17}$$

In Equation 14.16 (p. 612),

$$E_q=\ h_q-\frac{p}{\rho_q}+\frac{v^2}{2} \tag{14.18}$$

where h_q for each phase is based on the specific heat of that phase and the shared temperature.

The properties ρ, k_{eff} (effective thermal conductivity) and μ_{eff} (effective viscosity) are calculated by volumetric averaging over the phases. The source term, S_h, contains contributions from radiation, as well as any other volumetric heat sources.

As with the velocity field, the accuracy of the temperature near the interface is limited in cases where large temperature differences exist between the phases. Such problems also arise in cases where the properties vary by several orders of magnitude. For example, if a model includes liquid metal in combination with air, the conductivities of the materials can differ by as much as four orders of magnitude. Such large discrepancies in properties lead to equation sets with anisotropic coefficients, which in turn can lead to convergence and precision limitations.

14.3.8. Additional Scalar Equations

Depending upon your problem definition, additional scalar equations may be involved in your solution. In the case of turbulence quantities, a single set of transport equations is solved, and the turbulence

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

variables (for example, k and ε or the Reynolds stresses) are shared by the phases throughout the field.

14.3.9. Surface Tension and Adhesion

In Ansys Fluent, you can include the effects of surface tension along the interface between a pair of phases in your simulation. You can specify a surface tension coefficient as a constant, as a function of temperature through a polynomial, or as a function of any variable through a UDF. The solver will include the additional tangential stress terms (causing what is termed as Marangoni convection) that arise due to the variation in surface tension coefficient. Variable surface tension coefficient effects are usually important only in zero/near-zero gravity conditions.

Wall adhesion effects can be included by the additional specification of the contact angles between the phases and the walls, as well as at porous jumps.

14.3.9.1. Surface Tension

Surface tension arises as a result of attractive forces between molecules in a fluid. Consider an air bubble in water, for example. Within the bubble, the net force on a molecule due to its neighbors is zero. At the surface, however, the net force is radially inward, and the combined effect of the radial components of force across the entire spherical surface is to make the surface contract, thereby increasing the pressure on the concave side of the surface. The surface tension is a force, acting only at the surface, that is required to maintain equilibrium in such instances. It acts to balance the radially inward inter-molecular attractive force with the radially outward pressure gradient force across the surface. In regions where two fluids are separated, but one of them is not in the form of spherical bubbles, the surface tension acts to minimize free energy by decreasing the area of the interface.

In Ansys Fluent, two surface tension models exist: the continuum surface force (CSF) and the continuum surface stress (CSS). The two models are described in detail in the sections that follow.

Note:

The calculation of surface tension effects on triangular and tetrahedral meshes is not as accurate as on quadrilateral and hexahedral meshes. The region where surface tension effects are most important should therefore be meshed with quadrilaterals or hexahedra.

14.3.9.1.1. The Continuum Surface Force Model

The continuum surface force (CSF) model proposed by Brackbill et al. [74] (p. 1057) interprets surface tension as a continuous, three-dimensional effect across an interface, rather than as a boundary value condition on the interface. Surface tension effects are modeled by adding a source term in the momentum equation. To understand the origin of the source term, consider the special case where the surface tension is constant along the surface, and where only the forces normal to the interface are considered. It can be shown that the pressure drop across the surface depends upon the surface tension coefficient, σ, and the surface curvature as measured by two radii in orthogonal directions, R_1 and R_2:

$$p_2 - p_1 = \sigma \left(\frac{1}{R_1} + \frac{1}{R_2} \right) \tag{14.19}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

where p_1 and p_2 are the pressures in the two fluids on either side of the interface.

The surface curvature is computed from local gradients in the surface normal at the interface. Let n be the surface normal, defined as the gradient of α_q, the volume fraction of the q^{th} phase.

$$n = \nabla \alpha_q \tag{14.20}$$

The curvature, κ, is defined in terms of the divergence of the unit normal, $\hat{n}$ [74] (p. 1057):

$$\kappa = \nabla \cdot \hat{n} \tag{14.21}$$

where

$$\hat{n} = \frac{n}{|n|} \tag{14.22}$$

The surface tension can be written in terms of the pressure jump across the surface. The force at the surface can be expressed as a volume force using the divergence theorem. It is this volume force that is the source term that is added to the momentum equation. It has the following form:

$$F_{vol} = \sum_{pairs ij, i<j} \sigma_{ij} \frac{\alpha_i \rho_i \kappa_j \nabla \alpha_j + \alpha_j \rho_j \kappa_i \nabla \alpha_i}{\frac{1}{2}\left(\rho_i + \rho_j\right)} \tag{14.23}$$

This expression allows for a smooth superposition of forces near cells where more than two phases are present. If only two phases are present in a cell, then $\kappa_i = -\kappa_j$ and $\nabla \alpha_i = -\nabla \alpha_j$, and Equation 14.23 (p. 614) simplifies to

$$F_{vol} = \sigma_{ij} \frac{\rho \kappa_i \nabla \alpha_i}{\frac{1}{2}\left(\rho_i + \rho_j\right)} \tag{14.24}$$

where ρ is the volume-averaged density computed using Equation 14.14 (p. 611). Equation 14.24 (p. 614) shows that the surface tension source term for a cell is proportional to the average density in the cell.

14.3.9.1.2. The Continuum Surface Stress Model

The Continuum Surface Stress (CSS) method is an alternative way to modeling surface tension in a conservative manner, unlike the non-conservative formulation of the Continuum Surface Force (CSF) method. CSS avoids the explicit calculation of curvature, and could be represented as an anisotropic variant of modeling capillary forces based on surface stresses.

In the CSS method, the surface stress tensor due to surface tension is represented as

$$T = \sigma\left(I - \hat{n} \otimes \hat{n}\right) |\vec{n}| \tag{14.25}$$

$$\vec{n} = \nabla \alpha \tag{14.26}$$

$$\hat{n} = \frac{\vec{n}}{|\vec{n}|} \tag{14.27}$$

where,

I = unit tensor

σ = surface tension coefficient

$\otimes$ = tensor product of the two vectors: the original normal and the transformed normal

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

α = volume fraction

$\vec{n}$ = volume fraction gradient

$$T=\sigma\left(|\nabla\alpha|I-\frac{\nabla\alpha\otimes\nabla\alpha}{|\nabla\alpha|}\right) \tag{14.28}$$

The surface tension force is represented as

$$F_{CSS}=\nabla.T \tag{14.29}$$

14.3.9.1.3. Comparing the CSS and CSF Methods

The CSS method provides few added advantages over the CSF method, especially for cases involving variable surface tension. Both CSS and CSF methods introduce parasitic currents at the interface due to the imbalance of the pressure gradient and surface tension force.

In the CSF method, the surface tension force is represented in a nonconservative manner as follows:

$$F_{CSF}=\sigma\kappa\nabla\alpha \tag{14.30}$$

where κ is the curvature. This expression is valid only for constant surface tension.

For variable surface tension, the CSF formulation requires you to model an additional term in the tangential direction to the interface based on the surface tension gradient.

In the CSS method, surface tension force is represented in a conservative manner as follows:

$$F_{CSS}=\nabla.\left[\sigma\left(|\nabla\alpha|I-\frac{\nabla\alpha\otimes\nabla\alpha}{|\nabla\alpha|}\right)\right] \tag{14.31}$$

The CSS method does not require any explicit calculation for the curvature. Therefore, It performs more physically in under-resolved regions, such as sharp corners.

The CSS method does not require any additional terms for modeling variable surface tension due to its conservative formulation.

14.3.9.1.4. When Surface Tension Effects Are Important

The importance of surface tension effects is determined based on the value of two dimensionless quantities: the Reynolds number, Re, and the capillary number, Ca; or the Reynolds number, Re, and the Weber number, We. For Re $\ll 1$, the quantity of interest is the capillary number:

$$Ca=\frac{\mu U}{\sigma} \tag{14.32}$$

and for Re $\gg 1$, the quantity of interest is the Weber number:

$$We=\frac{\rho L U^2}{\sigma} \tag{14.33}$$

where U is the free-stream velocity. Surface tension effects can be neglected if $Ca \gg 1$ or $We \gg 1$.

To include the effects of surface tension in your model, refer to Including Surface Tension and Adhesion Effects in the User's Guide.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

14.3.9.2. Wall Adhesion

The effects of wall adhesion at fluid interfaces in contact with rigid boundaries in equilibrium can be estimated easily within the framework of the CSF model. Rather than impose this boundary condition at the wall itself, the contact angle that the fluid is assumed to make with the wall is used to adjust the surface normal in cells near the wall. This so-called dynamic boundary condition results in the adjustment of the curvature of the surface near the wall.

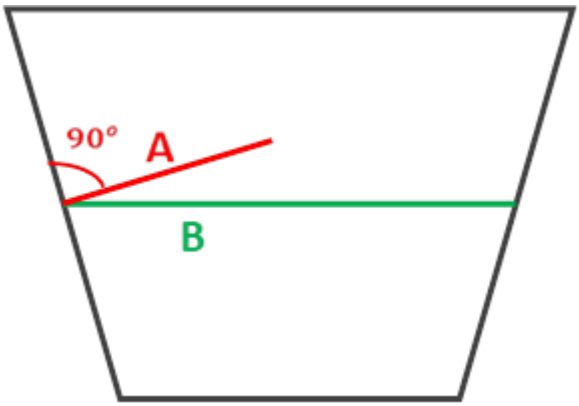
If θ_w is the contact angle at the wall, then the surface normal at the live cell next to the wall is

$$\hat{n}=\hat{n}_w\cos\theta_w+\hat{t}_w\sin\theta_w \tag{14.34}$$

where $\hat{n}_w$ and $\hat{t}_w$ are the unit vectors normal and tangential to the wall, respectively. The combination of this contact angle with the normally calculated surface normal one cell away from the wall determine the local curvature of the surface, and this curvature is used to adjust the body force term in the surface tension calculation.

Figure 14.3: Free Surface Positions With and Without Wall Adhesion (p. 616) shows the position of the free surface when wall adhesion is used with the contact angle of 90 degrees (**A**) and when wall adhesion is not used (**B**)

Figure 14.3: Free Surface Positions With and Without Wall Adhesion



If wall adhesion is used with the contact angle of 90 degrees (**A**), then the boundary condition of the contact angle forces the interface to be normal to the boundary in the curvature calculation.

If wall adhesion is not used (**B**), then for the curvature calculation, the volume fraction gradient at the wall is copied from the cells adjacent to the wall, and there is no boundary condition enforcement of the contact angle.

To include wall adhesion in your model, refer to Including Surface Tension and Adhesion Effects in the User's Guide.

14.3.9.3. Jump Adhesion

Similar to wall adhesion, there is also an option to provide jump adhesion when using the VOF model. Here, the contact angle treatment is applicable to each side of the porous jump boundary by assuming the same contact angle at both the sides.

Therefore, if θ_w is the contact angle at the porous jump, then the surface normal at the neighboring cell to the porous jump is

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$\hat{n}=\hat{n}_w\cos\theta_w+\hat{t}_w\sin\theta_w \tag{14.35}$$

where $\hat{n}_w$ and $\hat{t}_w$ are the unit vectors normal and tangential to the porous jump, respectively.

To include jump adhesion in your model, refer to Steps for Setting Boundary Conditions in the User's Guide.

Ansys Fluent provides two methods of jump adhesion treatment at the porous jump boundary:

- **Constrained Two-Sided Adhesion Treatment:**

  The constrained two-sided adhesion treatment option imposes constraints on the adhesion treatment. Here, the contact angle treatment will be applied to only the side(s) of the porous jump that is(are) adjacent to the non-porous fluid zones. Hence, the contact angle treatment will not be applied to the side(s) of the porous jump that is(are) adjacent to porous media zones. If the constrained two-sided adhesion treatment is disabled, the contact angle treatment will be applied to all sides of the porous jump.

  Note:

  This is the default treatment in Ansys Fluent.

- **Forced Two-Sided Adhesion:**

  Ansys Fluent allows you to use forced two-sided contact angle treatment for fluid zones without any imposed constraints. Refer to Steps for Setting Boundary Conditions in the User's Guide to learn how to apply this option.

14.3.10. Open Channel Flow

Ansys Fluent can model the effects of open channel flow (for example, rivers, dams, and surface-piercing structures in unbounded stream) using the VOF formulation and the open channel boundary condition. These flows involve the existence of a free surface between the flowing fluid and fluid above it (generally the atmosphere). In such cases, the wave propagation and free surface behavior becomes important. Flow is generally governed by the forces of gravity and inertia. This feature is mostly applicable to marine applications and the analysis of flows through drainage systems.

Open channel flows are characterized by the dimensionless Froude Number, which is defined as the ratio of inertia force and hydrostatic force.

$$Fr=\frac{V}{\sqrt{gy}} \tag{14.36}$$

where V is the velocity magnitude, g is gravity, and y is a length scale, in this case, the distance from the bottom of the channel to the free surface. The denominator in Equation 14.36 (p. 617) is the propagation speed of the wave. The wave speed as seen by the fixed observer is defined as

$$V_w=V\pm\sqrt{gy} \tag{14.37}$$

Based on the Froude number, open channel flows can be classified in the following three categories:

- When $Fr < 1$, that is, $V < \sqrt{gy}$ (therefore $V_w < 0$ or $V_w > 0$), the flow is known to be *subcritical* where disturbances can travel upstream as well as downstream. In this case, downstream conditions might affect the flow upstream.
- When $Fr = 1$ (therefore $V_w = 0$), the flow is known to be critical, where upstream propagating waves remain stationary. In this case, the character of the flow changes.
- When $Fr > 1$, that is, $V > \sqrt{gy}$ (therefore $V_w > 0$), the flow is known to be *supercritical* where disturbances cannot travel upstream. In this case, downstream conditions do not affect the flow upstream.

14.3.10.1. Upstream Boundary Conditions

There are two options available for the upstream boundary condition for open channel flows:

- pressure inlet
- mass flow rate

14.3.10.1.1. Pressure Inlet

The total pressure p_0 at the inlet can be given as

$$p_0 = \frac{1}{2}\rho V^2 + \left(\rho - \rho_0\right)|\vec{g}|\left(\hat{g}\cdot\left(\vec{b} - \vec{a}\right)\right) \tag{14.38}$$

where $\vec{b}$ and $\vec{a}$ are the position vectors of the face centroid and any point on the free surface, respectively, Here, the free surface is assumed to be horizontal and normal to the direction of gravity. $\vec{g}$ is the gravity vector, $|\vec{g}|$ is the gravity magnitude, $\hat{g}$ is the unit vector of gravity, V is the velocity magnitude, ρ is the density of the mixture in the cell, and ρ_0 is the reference density.

From this, the dynamic pressure q is

$$q = \frac{\rho}{2} V^2 \tag{14.39}$$

and the static pressure p_s is

$$p_s = \left(\rho - \rho_0\right)|\vec{g}|\left(\hat{g}\cdot\left(\vec{b} - \vec{a}\right)\right) \tag{14.40}$$

which can be further expanded to

$$p_s = \left(\rho - \rho_0\right)|\vec{g}|\left(\left(\hat{g}\cdot\vec{b}\right) + y_{local}\right) \tag{14.41}$$

where the distance from the free surface to the reference position, y_{local}, is

$$y_{local} = -\left(\vec{a}\cdot\hat{g}\right) \tag{14.42}$$

14.3.10.1.2. Mass Flow Rate

The mass flow rate for each phase associated with the open channel flow is defined by

$$\dot{m}_{phase} = \rho_{phase}\left(Area_{phase}\right)\left(Velocity\right) \tag{14.43}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

14.3.10.1.3. Volume Fraction Specification

In open channel flows, Ansys Fluent internally calculates the volume fraction based on the input parameters specified in the boundary conditions dialog box, therefore this option has been disabled.

For subcritical inlet flows (Fr < 1), Ansys Fluent reconstructs the volume fraction values on the boundary by using the values from the neighboring cells. This can be accomplished using the following procedure:

- Calculate the node values of volume fraction at the boundary using the cell values.
- Calculate the volume fraction at the each face of boundary using the interpolated node values.

For supercritical inlet flows (Fr > 1), the volume fraction value on the boundary can be calculated using the fixed height of the free surface from the bottom.

14.3.10.2. Downstream Boundary Conditions

14.3.10.2.1. Pressure Outlet

Determining the static pressure is dependent on the pressure specification method:

- Free Surface Level: The static pressure is dictated by Equation 14.40 (p. 618) and Equation 14.42 (p. 618).

  For subcritical outlet flows (Fr <1), the static pressure is taken from the pressure profile specified over the boundary, otherwise the pressure is taken from the neighboring cell. For supercritical flows (Fr > 1), the pressure is always taken from the neighboring cell.

- From Neighboring Cell: The static pressure is always taken from the neighboring cell.
- Gauge Pressure: The static pressure is a user-specified value.

14.3.10.2.2. Outflow Boundary

Outflow boundary conditions can be used at the outlet of open channel flows to model flow exits where the details of the flow velocity and pressure are not known prior to solving the flow problem. If the conditions are unknown at the outflow boundaries, then Ansys Fluent will extrapolate the required information from the interior.

It is important, however, to understand the limitations of this boundary type:

- You can only use single outflow boundaries at the outlet, which is achieved by setting the flow rate weighting to 1. In other words, outflow splitting is not permitted in open channel flows with outflow boundaries.
- There should be an initial flow field in the simulation to avoid convergence issues due to flow reversal at the outflow, which will result in an unreliable solution.
- An outflow boundary condition can only be used with mass-flow inlets. It is not compatible with pressure inlets and pressure outlets. For example, if you choose the inlet as **pressure-inlet**, then you can only use **pressure-outlet** at the outlet. If you choose the inlet as **mass-flow-inlet**,

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

then you can use either **outflow** or **pressure-outlet** boundary conditions at the outlet. Note that this only holds true for open channel flow.

- Note that the outflow boundary condition assumes that flow is fully developed in the direction perpendicular to the outflow boundary surface. Therefore, such surfaces should be placed accordingly.

14.3.10.2.3. Backflow Volume Fraction Specification

Ansys Fluent internally calculates the volume fraction values on the outlet boundary by using the neighboring cell values, therefore, this option is disabled.

14.3.10.3. Numerical Beach Treatment

In certain applications, it is desirable to suppress the numerical reflection caused by an outlet boundary for passing waves. To avoid wave reflection, a damping sink term is added in the momentum equation for the cell zone in the vicinity of the pressure outlet boundary [502] (p. 1082), [739] (p. 1096).

$$S=-[C_1\rho V+\frac{1}{2}C_2\rho|V|V]f(z)f(x) \tag{14.44}$$

where

$\hat{z}$ = vertical direction along gravity

$\hat{x}$ = flow direction

S = momentum sink term in the $\hat{z}$ direction

C_1 = linear damping resistance (1/s) (default value is 10)

C_2 = quadratic damping resistance (1/m) (default value is 10)

V = velocity along the $\hat{z}$ direction

z = distance from the free surface level

x = distance along the flow direction $\hat{x}$

$f(x)$ = damping function in the $\hat{x}$ direction

$f(z)$ = damping function in the $\hat{z}$ direction

The scaling factors in the $\hat{x}$ and $\hat{z}$ directions are defined by Equation 14.45 (p. 620) and Equation 14.46 (p. 620), respectively:

$$r_x=\frac{x-x_s}{x_e-x_s} \tag{14.45}$$

$$r_z=\frac{z-z_{fs}}{z_b-z_{fs}} \tag{14.46}$$

where the damping functions in the $\hat{x}$ and $\hat{z}$ directions, respectively, are

$$f(x)=(r_x)^2 \tag{14.47}$$

$$f(z)=1-r_z \tag{14.48}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

In Equation 14.45 (p. 620), x_s and x_e are the start and end points of the damping zone in the $\hat{x}$ direction. In Equation 14.46 (p. 620), z_{fs} and z_b are the free surface and bottom level along the $\hat{z}$ direction.

Note:

This option is available with Open Channel Flow (p. 617) and Open Channel Wave Boundary Conditions (p. 621).

To include numerical beach in your simulation, see Numerical Beach Treatment for Open Channels.

14.3.11. Open Channel Wave Boundary Conditions

Open channel wave boundary conditions allow you to simulate the propagation of regular/irregular waves, which is useful in the marine industry for analyzing wave kinematics and wave impact loads on moving bodies and offshore structures.

Small amplitude wave theories are generally applicable to lower wave steepness and lower relative height, while finite amplitude wave theories are more appropriate for either increasing wave steepness or increasing relative height. Wave steepness is generally defined as the ratio of wave height to wave length, and relative height is defined as the ratio of wave height to the liquid depth.

Ansys Fluent provides the following options for incoming surface gravity waves through velocity inlet boundary conditions:

- First order Airy wave theory, which is applied to small amplitude waves in shallow to deep liquid depth ranges, and is linear in nature.
- Higher order Stokes wave theories, which are applied to finite amplitude waves in intermediate to deep liquid depth ranges, and are nonlinear in nature.
- Higher order Cnoidal/Solitary wave theories, which are applied to finite amplitude waves in shallow depth ranges, and are nonlinear in nature.
- Superposition of linear waves, which is used to generate various physical phenomena such as interference, betas, standing waves, and irregular waves.
- Long/Short-crested wave spectrums, which are used for modeling nonlinear random waves in intermediate to deep liquid depth ranges based on a wave energy distribution function.

Short gravity wave expressions for each wave theory are based on infinite liquid height, whereas shallow or intermediate wave expressions are based on finite liquid height.

The wave height H is defined as follows:

$$H=2A=A_t+A_c \tag{14.49}$$

where A is the wave amplitude, A_t is the wave amplitude at the trough, and A_c is the wave amplitude at the crest. For linear wave theory, $A_t=A_c$ and for nonlinear wave theory, $A_t \neq A_c$.

The wave number (k) is:

$$k=\frac{2\pi}{\lambda} \tag{14.50}$$

where λ is the wave length.

The vector form for the wave number K is

$$\vec{K}=k_x\hat{x}+k_y\hat{y} \tag{14.51}$$

where $\hat{x}$ is the reference wave propagation direction, $\hat{z}$ is the direction opposite to gravity and $\hat{y}$ is the direction normal to $\hat{x}$ and $\hat{z}$. The wave numbers in the $\hat{x}$ and $\hat{y}$ directions are defined by Equation 14.52 (p. 622) and Equation 14.53 (p. 622), respectively:

$$k_x=k\cos\theta \tag{14.52}$$

$$k_y=k\sin\theta \tag{14.53}$$

θ is the wave heading angle, which is defined as the angle between the wavefront and reference wave propagation direction, in the plane of $\hat{x}$ and $\hat{y}$.

The effective wave frequency ω_e is defined as

$$\omega_e=\omega+\vec{K}\cdot\vec{U} \tag{14.54}$$

where ω is the wave frequency and $\vec{U}$ is the averaged velocity of the flow current. The effects of a moving object could also be incorporated with the flow current when the flow is specified relative to the motion of the moving object.

The wave speed or celerity c is defined as

$$c=\frac{\omega}{k} \tag{14.55}$$

The final velocity vector for incoming waves obtained by superposing all the velocity components is represented as:

$$\vec{V}=\vec{U}+u\hat{x}+v\hat{y}+w\hat{z} \tag{14.56}$$

where u, v, and w are the velocity components of the surface gravity wave in the $\hat{x}$, $\hat{y}$, and $\hat{z}$ directions, respectively.

The variable α is used in all of the wave theories and is defined as:

$$\alpha=k_xx+k_yy-\omega_et+\varepsilon \tag{14.57}$$

x and y are the space coordinates in the $\hat{x}$ and $\hat{y}$ directions, respectively, ε is the phase difference, and t is the time.

14.3.11.1. Airy Wave Theory

The wave profile for a linear wave is given as

$$\zeta\left(X,t\right)=A\cos\alpha \tag{14.58}$$

where α is as define as above in Open Channel Wave Boundary Conditions (p. 621).

The wave frequency ω is defined as follows for shallow/intermediate waves:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$\omega=\sqrt{gk\tanh\left(kh\right)} \tag{14.59}$$

and as follows for short gravity waves:

$$\omega=\sqrt{gk} \tag{14.60}$$

where h is the liquid height, k is the wave number, and g is the gravity magnitude.

The velocity components for the incident wave boundary condition can be described in terms of shallow/intermediate waves and short gravity waves.

Velocity Components for Shallow/Intermediate Waves

$$\begin{pmatrix}u\\v\end{pmatrix}=\frac{gkA}{\omega}\frac{\cosh\left[k(z+h)\right]}{\cosh(kh)}\begin{pmatrix}\cos\theta\\\sin\theta\end{pmatrix}\cos\alpha \tag{14.61}$$

$$w=\frac{gkA}{\omega}\frac{\sinh\left[k(z+h)\right]}{\cosh(kh)}\sin\alpha \tag{14.62}$$

Velocity Components for Short Gravity Waves

$$\begin{pmatrix}u\\v\end{pmatrix}=\frac{gkA}{\omega}e^{kz}\begin{pmatrix}\cos\theta\\\sin\theta\end{pmatrix}\cos\alpha \tag{14.63}$$

$$w=\frac{gkA}{\omega}e^{kz}\sin\alpha \tag{14.64}$$

where z is the height from the free surface level in the direction $\hat{z}$, which is opposite to that of gravity. For more information on how to use and set up this model, refer to Modeling Open Channel Wave Boundary Conditions in the User's Guide.

14.3.11.2. Stokes Wave Theories

Fluent formulates the Stokes wave theories based on the work by John D. Fenton[174] (p. 1063). These wave theories are valid for high steepness finite amplitudes waves operating in intermediate to deep liquid depth range.

The generalized expression for wave profiles for higher order Stokes theories (second to fifth order) is given as

$$\zeta\left(X,t\right)=\frac{1}{k}\sum_{i=1}^{n}\sum_{j=1}^{i}b_{ij}\xi^{i}\cos(j\alpha) \tag{14.65}$$

The generalized expression for the associated velocity potential is given as

$$\Phi\left(X,t\right)=\frac{1}{k}\sqrt{\frac{g}{k}tanh(kh)}\sum_{i=1}^{n}\xi^{i}\sum_{j=1}^{i}a_{ij}\cosh(\mathrm{jk}\left(\mathrm{z+h}\right))\sin(j\alpha) \tag{14.66}$$

Where,

$$\xi=\frac{kH}{2} \tag{14.67}$$

ξ is referred to as wave steepness.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

n = wave theory index (2 to 5: From 2nd order Stokes to 5th order Stokes respectively).

Wave celerity, c is given as

$$c= \sqrt{\frac{g}{k}tanh(kh)}\ \left(1+\ c_3\xi^2+c_5\xi^4\right) \tag{14.68}$$

For 2nd order Stokes, c_3=c_5=0. (Therefore, 2nd order Stokes uses the same dispersion relation as used by first order wave theory.) For 3rd and 4th order Stokes, c_5=0.

a_{ij} , b_{ij}, C_i = complex expressions of kh as discussed in [174] (p. 1063).

Wave frequency, ω, is defined as

$$\omega=kc \tag{14.69}$$

Velocity components for surface gravity waves are derived from the velocity potential function.

$$u=\frac{\partial \Phi}{\partial x}\cos\theta \tag{14.70}$$

$$v=\frac{\partial \Phi}{\partial x}\sin\theta \tag{14.71}$$

$$w=\frac{\partial \Phi}{\partial z} \tag{14.72}$$

14.3.11.3. Cnoidal/Solitary Wave Theory

Fluent formulates the Cnoidal/Solitary wave theories that are expressed using complex Jacobian and elliptic functions based on the work by John D. Fenton (1998) [175] (p. 1063). The cnoidal solution displays long flat troughs and narrow crests of real waves in shallow waters. In the limit of infinite wavelength, the cnoidal solution describes a solitary wave as a wave with a single hump, having no troughs. Due to the complexity of the cnoidal wave theories, solitary wave theories are more widely used for shallow depth regimes.

Note:

For simplicity, the higher order terms, which are functions of relative height, have been omitted from the numerical details below. Refer to the work of John D. Fenton [175] (p. 1063) for detailed numerical expressions.

The elliptic function parameter, m, is calculated by solving the following nonlinear equation:

$$\frac{\lambda}{h}=4K(m)\sqrt{\frac{3H}{h}}\ [1+Higher\cdot order\cdot terms] \tag{14.73}$$

Trough height from the bottom, h_{tr}, is derived from the following relationship:

$$\frac{h_{tr}}{h}=1-\frac{H}{h}\frac{E(m)}{K(m)}+Higher\cdot order\cdot terms \tag{14.74}$$

Wave number, k, is defined as:

$$k=\frac{1}{h_{tr}}\sqrt{\frac{3H}{4h_{tr}}}\ \left[1+Higher\cdot order\cdot terms\right] \tag{14.75}$$

Wave celerity, c, is defined as:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$c=\sqrt{gh_{tr}}\left(1+\frac{H}{h}\left[\frac{1}{2}-\frac{E(m)}{K(m)}\right]+Higher\cdot order\cdot terms\right) \tag{14.76}$$

Wave frequency, ω, is defined as:

$$\omega=kc \tag{14.77}$$

The wave profile for a shallow wave is defined as:

$$\zeta(X,t)=-h+h_{tr}\left[1+\frac{H}{h_{tr}}cn^2(\alpha,m)+Higher\cdot order\cdot terms\right] \tag{14.78}$$

Where α is defined as:

$$\alpha=k_x(x-x_0)+k_y y-\omega_e t \tag{14.79}$$

where x_0 is the translational distance from reference frame origin in the $\hat{x}$ direction.

Velocity components for the fifth order wave theories are expressed as:

$$u=\left[c-\sqrt{gh_{tr}}\left\{1-\sum_{i=1}^{5}\left(\frac{4(kh_{tr})^2}{3}\right)^i\sum_{j=0}^{i-1}\left(\frac{z+h}{h_{tr}}\right)^{2j}\sum_{l=0}^{i}cn^{2l}(\alpha,m)C_{ijl}\right\}\right]\cos(\theta) \tag{14.80}$$

$$v=\left[c-\sqrt{gh_{tr}}\left\{1-\sum_{i=1}^{5}\left(\frac{4(kh_{tr})^2}{3}\right)^i\sum_{j=0}^{i-1}\left(\frac{z+h}{h_{tr}}\right)^{2j}\sum_{l=0}^{i}cn^{2l}(\alpha,m)C_{ijl}\right\}\right]\sin(\theta) \tag{14.81}$$

$$\begin{aligned} w=\ & \sqrt{gh_{tr}}\,2kh_{tr}cn(\alpha,m)sn(\alpha,m)dn(\alpha,m) \\ & \left[\sum_{i=1}^{5}\left(\frac{4(kh_{tr})^2}{3}\right)^i\sum_{j=0}^{i-1}\left(\frac{z+h}{h_{tr}}\right)^{2j+1}\sum_{l=1}^{i}cn^{2(l-1)}(\alpha,m)\frac{1}{2j+1}C_{ijl}\right] \end{aligned} \tag{14.82}$$

where C_{ijl} are the numerical coefficients mentioned in the reference [175] (p. 1063).

Note:

Solitary wave theory expressions are derived by assuming that the waves have infinite wave length. Based on this assumption we get:

$$m=1$$

$$h_{tr}=h$$

14.3.11.4. Choosing a Wave Theory

Applying linear versus nonlinear wave theories to shallow waves is limited by the Ursell number. For nonlinear waves, the Ursell number criterion is based on waves being single-crested, without producing any secondary crests at the trough. Wave theories should be chosen in accordance with wave steepness and relative height, as waves try to acquire a nonlinear pattern with increasing wave steepness or relative height. Second and fourth order wave theories are more prone to produce secondary crests. Choosing the correct wave theory for a particular application depends on the input parameters within the wave breaking and stability limit, as described below:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

- **Mandatory Check for Complete Wave Regime within Wave Breaking Limit**

  The wave height to liquid depth ratio H/h (relative height) within the wave breaking limit is defined as:

  $$\left[\frac{H}{h}\right]_{\max,theoretical}=0.78 \tag{14.83}$$

  $$\left[\frac{H}{h}\right]_{\max,Numerical}=0.55 \tag{14.84}$$

  The wave height to wave length depth ratio H/λ (wave steepness) within the breaking limit is defined as:

  $$\left[\frac{H}{\lambda}\right]_{\max,theoretical}=0.142 \tag{14.85}$$

  $$\left[\frac{H}{\lambda}\right]_{\max,Numerical}=0.12 \tag{14.86}$$

- **Check for Wave Theories within Wave Breaking and Stability Limit**

  For this check, the wave type is represented with the appropriate wave theory in the following manner:

  – **Linear Wave**: Airy Wave Theory

  – **Stokes Wave**: Fifth Order Stokes Wave Theory

  – **Shallow Wave**: Fifth Order Cnoidal/Solitary Wave Theory

  Checks for wave theories between second to fourth order are performed between linear and fifth order Stokes wave.

  – **Wave Regime Check**

    The liquid height to wave length ratio h/λ for short gravity waves is defined as:

    $$\left[\frac{h}{\lambda}\right]_{\min}=0.5 \tag{14.87}$$

    For Stokes waves the ratio is defined as:

    $$\left[\frac{h}{\lambda}\right]_{\min}=0.06 \tag{14.88}$$

    For shallow waves, the ratio is defined as:

    $$\left[\frac{h}{\lambda}\right]_{\max}=0.085 \tag{14.89}$$

  – **Wave Steepness Check**

    The wave height to wave length ratio H/λ for linear waves is defined as:

    $$\left[\frac{H}{\lambda}\right]_{\max}=0.02\tanh\left(2\pi\frac{h}{\lambda}\right) \tag{14.90}$$

    For Stokes waves the ratio is defined as:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$\left[\frac{H}{\lambda}\right]_{\max}=0.142\tanh\left(2\pi\frac{h}{\lambda}\right) \tag{14.91}$$

Note:

Stokes waves are generally stable for a wave steepness below 0.1. Waves may be either stable or unstable in the range of 0.1 to 0.12. Waves are most likely to break at wave steepness values above 0.12.

For shallow waves, the ratio is defined as:

$$\left[\frac{H}{\lambda}\right]_{\max}=0.142\left(2\pi\frac{h}{\lambda}\right) \tag{14.92}$$

– **Relative Height Check**

The wave height to liquid depth ratio H/h for a linear wave is defined as:

$$\left[\frac{H}{h}\right]_{\max}=0.1 \tag{14.93}$$

For Stokes wave in shallow regime the ratio is defined as:

$$\left[\frac{H}{h}\right]_{\max}=0.5 \tag{14.94}$$

Note:

Stokes waves are generally stable for a relative height below 0.4. Waves are most likely to break at a relative height above 0.4.

For shallow waves, the ratio is defined as:

$$\left[\frac{H}{h}\right]_{\max}=0.55 \tag{14.95}$$

– **Ursell Number Stability Criterion**

The Ursell number is defined as:

$$Ur=\frac{H\lambda^2}{h^3} \tag{14.96}$$

The Ursell number stability criterion for a linear wave is defined as:

$$\left(Ur\right)_{\max}=\frac{32\pi^2}{3} \tag{14.97}$$

The criterion for Stokes wave in shallow regime is defined as:

$$\left(Ur\right)_{\max}=\frac{8\pi^2}{3} \tag{14.98}$$

Note:

Stokes waves are generally stable for $Ur<10$.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Ursell numbers between 10 and 26 represent the transition to a shallow regime, therefore Stokes waves could be unstable. Stokes waves are more applicable in intermediate and deep liquid depth regimes.

The criterion for shallow waves is defined as:

$$\left(Ur\right)_{\min}=\frac{8\pi^2}{3} \tag{14.99}$$

14.3.11.5. Superposition of Waves

The principle of superposition of waves can be applied to two or more waves simultaneously passing through the same medium. The resultant wave pattern is produced by the summation of wave profiles and velocity potentials of the individual waves. Depending on the nature of the superposition, the following phenomena are possible:

- Interference: Superposition of waves of the same frequency and wave heights, travelling in the same direction. Constructive interference occurs in cases of waves travelling in the same phase, while destructive interference occurs in cases of waves travelling in opposite phases.
- Beats/Wave Group: Superposition of waves of same wave heights travelling in the same direction with nearly equal frequencies. The beats are generated by increase/decrease in the resultant frequency of the waves over time.

  For marine applications, this concept of wave superposition is known as a Wave Group. In a wave group, packets (group) of waves move with a group velocity that is slower than the phase speed of each individual wave within the packet.
- Stationary/Standing Waves: Superposition of identical waves travelling in opposite directions. This superposition forms a stationary wave where each particle executes simple harmonic motion with the same frequency but different amplitudes.
- Irregular Waves: Superposition of waves with different wave heights, frequencies, phase differences, and direction of propagation. Irregular sea waves are quite common in marine applications, and can be modeled by superimposing multiple waves.

Note:

The superposition principle is only valid for linear and small-amplitude waves. Superposition of finite amplitude waves may be approximate or invalid because of their nonlinear behavior and dependency of the wave dispersion on wave height.

14.3.11.6. Modeling of Random Waves Using Wave Spectrum

Random waves on the sea surface are generated by the action of the wind over the free surface. The higher the wind speed, the longer the wind duration, and the longer the distance over which the wind blows (fetch), the larger the resulting waves. You can simulate random wave action for open channel wave boundary conditions by specifying a wave spectrum that describes the distribution of energy over a range of wave frequencies. Fluent provides several formulations for wave spectra based on wind speed, direction, duration, and fetch.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

14.3.11.6.1. Definitions

Fully Developed Sea State

When the wind blown over the sea imparts its maximum energy to the waves, the sea is said to be fully developed. This means that the sea state is independent of the distance over which the wind blows (fetch) and the duration of the wind. Sea elevation can be assumed as statistically stable in this case.

The sea state is often characterized by the following parameters which can be estimated from the wind speed and fetch:

Significant Wave Height (H_s)

Mean wave height of the largest 1/3 of waves.

Peak Wave Frequency (ω_p) or Period

The frequency or period corresponding to the highest wave energy.

Long-Crested Sea

If the irregularity of the observed waves are only in the dominant wind direction, such that there are mainly unidirectional waves with different amplitudes but parallel to each other, the sea is referred to as a long-crested irregular sea.

Short-Crested Sea

When irregularities are apparent along the wave crests in multiple directions, the sea is referred to as short-crested.

14.3.11.6.2. Wave Spectrum Implementation Theory

14.3.11.6.2.1. Long-Crested Random Waves (Unidirectional)

Long-crested random wave spectrums are unidirectional and are a function of frequency only.

Ansys Fluent provides the following wave frequency spectrum formulations:

14.3.11.6.2.1.1. Pierson-Moskowitz Spectrum

14.3.11.6.2.1.2. JONSWAP Spectrum

14.3.11.6.2.1.3. TMA Spectrum

14.3.11.6.2.1.1. Pierson-Moskowitz Spectrum

The Pierson-Moskowitz spectrum is valid for a fully developed sea and assumes that waves come in equilibrium with the wind over an unlimited fetch [3] (p. 1053).

$$S_{PM}(\omega)=\frac{5}{16}\frac{H_s^2\omega_p^4}{\omega^5}e^{-\left(\frac{5\omega_p^4}{4\omega^4}\right)} \tag{14.100}$$

where:

ω = Wave frequency

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

ω_p = Peak wave frequency

H_s = Significant Wave Height

14.3.11.6.2.1.2. JONSWAP Spectrum

The JONSWAP spectrum is a fetch-limited version of the Pierson-Moskowitz spectrum in which the wave spectrum is considered to continue to develop due to nonlinear wave-wave interactions for a very long time and distance. The peak in the wave spectrum is more pronounced compared to the Pierson-Moskowitz spectrum [3] (p. 1053).

$$S_{JS}(\omega)=S_{PM}(\omega)A_\gamma\gamma\text{^}\exp\left(-0.5\left(\frac{\omega-\omega_p}{\sigma\omega_p}\right)^2\right) \tag{14.101}$$

where:

$S_{\mathrm{PM}}(\omega)$ = Pierson-Moskowitz spectrum

ω = Wave frequency

ω_p = Peak wave frequency

γ = Peak shape parameter

$A_\gamma=1-0.287\ln(\gamma)$

σ= Spectral width parameter

$$\sigma=0.07 \quad \omega\leq\omega_p$$
$$\sigma=0.09 \quad \omega>\omega_p$$

14.3.11.6.2.1.3. TMA Spectrum

The TMA spectrum is a modified version of the JONSWAP wave spectrum that is valid for finite water depth [3] (p. 1053).

$$S_{TMA}(\omega)=S_{JS}(\omega)\phi(\omega,h) \tag{14.102}$$

$\phi(\omega,h)$ is a depth function given by:

$$\phi(\omega,h)=\frac{(\cosh(kh))^2}{(\sinh(kh))^2+\frac{\omega^2h}{g}} \tag{14.103}$$

k is the wave number calculated from the dispersion relation

$$\omega^2=gk\tanh(kh)$$

where g is the magnitude of gravity and h is the liquid depth.

14.3.11.6.2.2. Short-Crested Random Waves (Multi-Directional)

Short-crested random waves are multidirectional and are a function of both frequency and direction. The short-crested wave spectrum is expressed as:

$$S(\omega,\theta)=S(\omega)G(\theta,\omega) \tag{14.104}$$

where:

$S(\omega)$ = Frequency spectrum

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$G(\theta,\omega)$ = Directional spreading function

The following equation must be satisfied:

$$\int_{\theta_{\min}}^{\theta_{\max}} S(\omega,\theta)d\theta = S(\omega)$$

which imposes the following condition on the spreading function:

$$\int_{\theta_{\min}}^{\theta_{\max}} G(\theta,\omega)d\theta = 1$$

where:

$\theta_{\min} = \theta_p - \theta_s$

$\theta_{\max} = \theta_p + \theta_s$

θ_p = Principal (mean) wave heading angle

θ_s = Angular spread from θ_p

$0 \le \theta_s \le \pi$

Ansys Fluent offers two options for specifying the directional spreading function, $G(\theta,\omega)$

14.3.11.6.2.2.1. Cosine-2s Power Function (Frequency Independent)

14.3.11.6.2.2.2. Hyperbolic Function (Frequency Dependent)

14.3.11.6.2.2.1. Cosine-2s Power Function (Frequency Independent)

In the cosine-2s power function formulation, the directional spreading function, $G(\theta,\omega)$, is frequency independent and is given by:

$$G(\theta,\omega) \approx G(\theta) = G(s)\cos^{2s}\left(\frac{\pi}{2\theta_s}(\theta-\theta_p)\right) \tag{14.105}$$

where:

$|\theta-\theta_p| \le \theta_s \quad 0 \le \theta_s \le \pi$

s = positive integer between 1 and n

$$G(s) = \left[\int_{-\theta_s}^{\theta_s} \cos^{2s}\left(\frac{\pi}{2\theta_s}\theta\right)d\theta\right]^{-1}$$

14.3.11.6.2.2.2. Hyperbolic Function (Frequency Dependent)

In the hyperbolic function formulation, the directional spreading function, $G(\theta,\omega)$, is frequency dependent and is given by:

$$G(\theta,\omega) = \frac{\beta}{2}\operatorname{sech}^2(\beta(\theta-\theta_p)) \tag{14.106}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$\beta=\begin{cases}2.61\left(\frac{\omega}{\omega_p}\right)^{1.3} & 0.56<\frac{\omega}{\omega_p}\leq 0.95\\ 2.28\left(\frac{\omega}{\omega_p}\right)^{-1.3} & 0.95<\frac{\omega}{\omega_p}\leq 1.60\\ 10^{y} & \frac{\omega}{\omega_p}>1.60\end{cases}$$

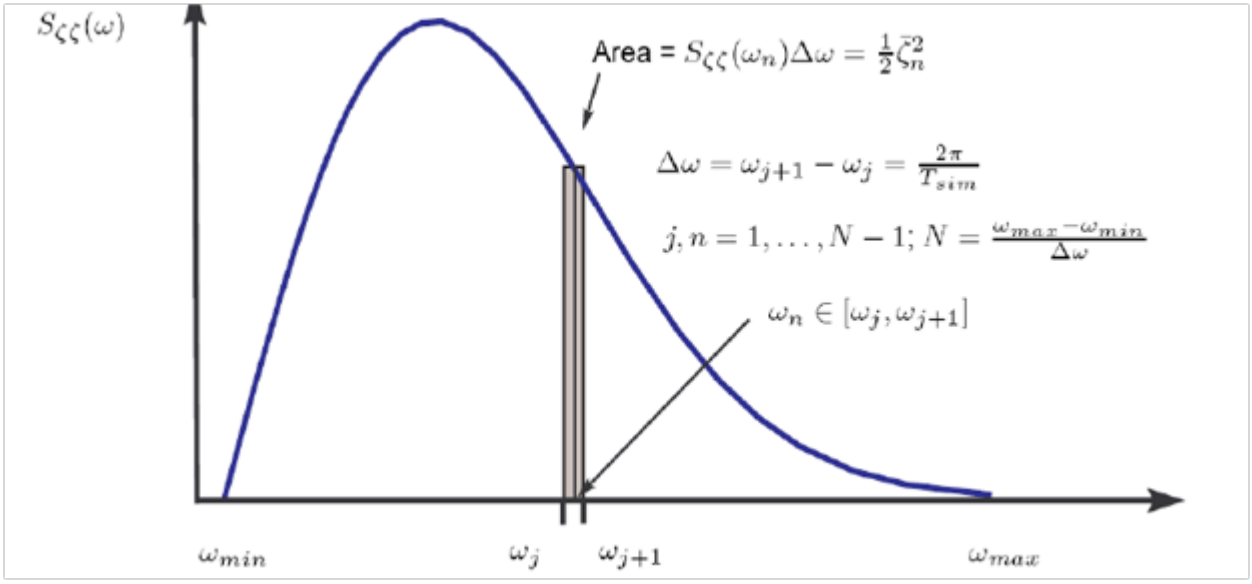
in which:

$$y=-0.4+0.8393e^{-0.567\ln\left(\frac{\omega}{\omega_p}\right)^2}$$

14.3.11.6.2.3. Superposition of Individual Wave Components Using the Wave Spectrum

Series of individual wave components are generated for the given frequency range and angular spread. These are then superposed using linear wave theory.

Figure 14.4: Typical Wave Spectrum



For each wave component, the amplitude is calculated from:

$$\frac{A^2}{2}=\int_{\theta-\Delta\theta/2}^{\theta+\Delta\theta/2}\int_{\omega-\Delta\omega/2}^{\omega+\Delta\omega/2} S(\omega,\theta)\,d\omega\,d\theta \tag{14.107}$$

The final wave profile after superposing all the individual wave components is given by:

$$\zeta(X,t)=\sum_{n=1}^{N}\sum_{m=1}^{M} A_{nm}\cos\alpha_{nm} \tag{14.108}$$

where:

N = Number of frequency components

M = Number of angular components

A_{nm} = Amplitude of individual wave component

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$\alpha_{nm}=\left(k_x\right)_{nm}x+\left(k_y\right)_{nm}y-\left(\omega_e\right)_n t+\varepsilon_{nm}$$

$$\left(k_x\right)_{nm}=k_n\cos\theta_m$$

$$\left(k_y\right)_{nm}=k_n\sin\theta_m$$

k_n = Wave number for nth frequency component

θ_m = Heading angle for the mth angular component

ε_{nm} = Random phase difference uniformly distributed between 0 and 2π

14.3.11.6.3. Choosing a Wave Spectrum and Inputs

The wave spectrum is represented by a wave energy distribution function characterized by two basic inputs: Significant Wave Height (H_s) and Peak Angular Frequency (ω_p). In addition, you will enter the minimum and maximum wave frequencies (ω_{min} and ω_{max}) and the number of frequencies at which to evaluate the spectrum. You should select the frequency range in such a way that most of the wave energy is contained in the specified range. In general, the recommended frequency range is:

Minimum Angular Frequency, $\omega_{min}=0.5\omega_p$

Maximum Angular Frequency, $\omega_{max}=2.5\omega_p$

When choosing the spectrum and input values, it is important to analyze and consider various parameters of the resulting waves. Fluent can compute various metrics based on your inputs and assess whether the chosen spectrum and parameters are appropriate. The metrics that are computed are detailed below.

Estimation of Wave Lengths

The wave length is given by $\lambda=\frac{2\pi}{k}$

where k is the wave number which is calculated from the following relationships:

For intermediate depth: $\omega^2=gk\tanh\left(kh\right)$,

For short gravity waves: $\omega^2=gk$,

Thus,

Minimum wave length: $\lambda_{min}=\frac{2\pi}{k_{min}}$

Maximum wave length: $\lambda_{max}=\frac{2\pi}{k_{max}}$

Peak wave length: $\lambda_p=\frac{2\pi}{k_p}$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Estimation of Time Periods

The peak time period is given by $T_p=\frac{2\pi}{\omega_p}$

The mean time period and the zero-upcrossing time periods are approximated as follows.

Mean time period: $T_m=0.7303+0.04936\gamma-0.006556\gamma^2+0.000361\gamma^3$

Zero upcrossing period: $T_z=0.6673+0.05037\gamma-0.00623\gamma^2+0.0003341\gamma^3$

where γ is the peak shape parameter.

Wave Regime Check

For the short gravity wave assumption, you should ensure that:

$$\frac{h}{\lambda_{max}}>0.5$$

In general, the TMA spectrum is appropriate for intermediate depth, while the JONSWAP and Pierson-Moskowitz spectrums are valid under the short gravity wave assumption.

Relative Height Check

For the intermediate/deep treatment to be valid, the relative height should satisfy:

$$\frac{H_s}{h}<0.25$$

Sea Steepness Check

The sea steepness is computed in two ways:

based on peak time period $S_p=\frac{2\pi H_s}{gT_p^2}$

based on zero-upcrossing time period $S_z=\frac{2\pi H_s}{gT_z^2}$

and compared with limiting relations for each method:

$$S_z\leq\begin{cases}\frac{1}{10} & T_z\leq 6s\\ \frac{1}{10}\frac{12-T_z}{6}+\frac{1}{15}\frac{T_z-6}{6} & 6s<T_z<12s\\ \frac{1}{15} & T_z\geq 12s\end{cases}$$

$$S_p\leq\begin{cases}\frac{1}{15} & T_p\leq 8s\\ \frac{1}{15}\frac{15-T_p}{7}+\frac{1}{25}\frac{T_p-8}{7} & 8s<T_p<15s\\ \frac{1}{25} & T_p\geq 15s\end{cases}$$

Estimation of Peak Shape Parameter

The peak shape parameter, γ is estimated by:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$\gamma = \begin{cases} 5 & \frac{T_p}{\sqrt{H_s}} \leq 3.6 \\ 5\exp\left(5.75 - 1.15\frac{T_p}{\sqrt{H_s}}\right) & 3.6 < \frac{T_p}{\sqrt{H_s}} < 5 \\ 1 & \frac{T_p}{\sqrt{H_s}} \geq 5 \end{cases}$$

Note:

For $\gamma = 1$, the JONSWAP spectrum reduces to the Pierson-Moskowitz spectrum.

14.3.11.7. Nomenclature for Open Channel Waves

Symbol	Description
$\hat{x}$	Reference wave propagation direction
$\hat{y}$	direction normal to $\hat{x}$ and $\hat{z}$
$\hat{z}$	Direction opposite to gravity
H	Wave Height
H_s	Significant Wave Height
ξ	Wave Steepness
A	Wave Amplitude
h	Liquid Height
h_{tr}	Trough Height
γ	Peak Shape Parameter
λ	Wave Length
θ	Wave Heading Angle
k	Wave Number
k_x	Wave Number in the $\hat{x}$ direction
k_y	Wave Number in the $\hat{y}$ direction
c	Wave Celerity
ω	Wave Frequency
ω_e	Effective Wave Frequency
ω_p	Peak Wave Frequency
m	Elliptical Function Parameter
$K(m)$	Complete Elliptical Integral of the First Kind
$E(m)$	Complete Elliptical Integral of the Second Kind

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Symbol	Description
$cn(\alpha,m), sn(\alpha,m), dn(\alpha,m)$	Jacobian Elliptic Functions
x	Spatial coordinate in the $\hat{x}$ direction
y	Spatial coordinate in the $\hat{y}$ direction
z	Spatial coordinate in the $\hat{z}$ direction
x_0	Translational distance from reference frame origin in the $\hat{x}$ direction
Φ	Velocity Potential
$\vec{U}$	Averaged velocity of flow current
u,v,w	Velocity components for surface gravity waves in the $\hat{x}$, $\hat{y}$, and $\hat{z}$ directions, respectively.

14.3.12. Coupled Level-Set and VOF Model

The level-set method is a popular interface-tracking method for computing two-phase flows with topologically complex interfaces. This is similar to the interface tracking method of the VOF model. In the level-set method [497] (p. 1082), the interface is captured and tracked by the level-set function, defined as a signed distance from the interface. Because the level-set function is smooth and continuous, its spatial gradients can be accurately calculated. This in turn will produce accurate estimates of interface curvature and surface tension force caused by the curvature. However, the level-set method is found to have a deficiency in preserving volume conservation [485] (p. 1081). On the other hand, the VOF method is naturally volume-conserved, as it computes and tracks the volume fraction of a particular phase in each cell rather than the interface itself. The weakness of the VOF method lies in the calculation of its spatial derivatives, since the VOF function (the volume fraction of a particular phase) is discontinuous across the interface. To overcome the deficiencies of the level-set method and the VOF method, a coupled level-set and VOF approach is provided in Ansys Fluent.

Important:

The coupled level-set and VOF model is specifically designed for two-phase flows, where no mass transfer is involved and is applied to transient flow problems only. The mesh should be restricted to quadrilateral, triangular or a combination of both for 2D cases and restricted to hexahedral, tetrahedral or a combination of both for 3D cases.

To learn how to use this model, refer to Including Coupled Level Set with the VOF Model.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

14.3.12.1. Theory

The level-set function φ is defined as a signed distance to the interface. Accordingly, the interface is the zero level-set, $\varphi(x,t)$ and can be expressed as $\Gamma=\{x|\varphi(x,t)=0\}$ in a two-phase flow system:

$$\varphi(x,t)=\begin{cases}+|d| & \text{if } x\in\text{the primary phase}\\ 0 & \text{if } x\in\Gamma\\ -|d| & \text{if } x\in\text{the secondary phase}\end{cases} \tag{14.109}$$

where d is the distance from the interface.

The evolution of the level-set function can be given in a similar fashion as to the VOF model:

$$\frac{\partial\varphi}{\partial t}+\nabla\cdot(\vec{u}\varphi)=0 \tag{14.110}$$

Where $\vec{u}$ is the underlying velocity field.

And the momentum equation can be written as

$$\frac{\partial(\rho\vec{u})}{\partial t}+\nabla\cdot(\rho\vec{u}\vec{u})=-\nabla p+\nabla\cdot\mu\left[\nabla\vec{u}+(\nabla\vec{u})^{T}\right]-\vec{F}_{sf}+\rho\vec{g} \tag{14.111}$$

14.3.12.1.1. Surface Tension Force

In Equation 14.111 (p. 637), $\vec{F}_{sf}$ is the force arising from surface tension effects given by:

$$\vec{F}_{sf}=\sigma\kappa\delta(\varphi)\vec{n} \tag{14.112}$$

where:

σ = surface tension coefficient

κ = local mean interface curvature

$\vec{n}$ = local interface normal

and

$$\delta(\varphi)=\begin{cases}0 & |\varphi|\geq\alpha\\ \frac{1+\cos(\pi\varphi/\alpha)}{2\alpha} & |\varphi|<\alpha\end{cases} \tag{14.113}$$

where a is the thickness of the interface and h is the grid spacing.

The normal, $\vec{n}$, and curvature, κ, of the interface can be estimated as

$$\vec{n}=\left.\frac{\nabla\varphi}{|\nabla\varphi|}\right|_{\varphi=0} \tag{14.114}$$

$$\kappa=\nabla\cdot\left.\frac{\nabla\varphi}{|\nabla\varphi|}\right|_{\varphi=0} \tag{14.115}$$

In some cases, applying the default surface tension force as written in Equation 14.112 (p. 637) can lead to spurious currents appearing in the solution. To mitigate these effects, Fluent offers two weighting functions that redistribute the surface tension force towards the heavier phase in the interface cells.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Density Correction

In the density correction formulation, Equation 14.112 (p. 637) is modified by introducing a density ratio:

$$\vec{F}_{sf}=\frac{\rho}{0.5(\rho_1+\rho_2)}\sigma\kappa\delta(\varphi)\vec{n} \tag{14.116}$$

where ρ is the volume-based density.

Heaviside Function Scaling

In the Heaviside function scaling formulation, Equation 14.112 (p. 637) is modified by introducing the Heaviside function:

$$\vec{F}_{sf}=2H_{\varphi}\sigma\kappa\delta(\varphi)\vec{n} \tag{14.117}$$

where:

$$H_{\varphi}=\begin{cases}0 & |\varphi|>a \quad (gas\ phase)\\ 1 & |\varphi|>a \quad (liquid\ phase)\\ \frac{1}{2}\left[1+\frac{\varphi}{a}+\frac{1}{\pi}\sin\left(\frac{\pi\varphi}{a}\right)\right] & |\varphi|\le a\end{cases} \tag{14.118}$$

and a is the thickness of the interface.

14.3.12.1.2. Re-initialization of the Level-set Function via the Geometrical Method

By nature of the transport equation of the level-set function Equation 14.110 (p. 637), it is unlikely that the distance constraint of $|\nabla\varphi|=1$ is maintained after its solution. The reasons for the lack of maintenance is due to the deformation of the interface, uneven profile, and thickness across the interface. Those errors will accumulate during the iteration process and cause large errors in mass and momentum solutions. A re-initialization process is therefore required for each time step. The geometrical interface-front construction method is used here. The geometrical method involves a simple concept and is reliable in producing accurate geometrical data for the interface front. The values of the VOF and the level-set function are both used to reconstruct the interface-front. Namely, the VOF model provides the size of the cut in the cell where the likely interface passes through, and the gradient of the level-set function determines the direction of the interface.

The concept of piecewise linear interface construction (PLIC) is also employed to construct the interface-front. The procedure for the interface–front reconstruction can be briefly described as follows and as shown in Figure 14.5: Schematic View of the Interface Cut Through the Front Cell (p. 639):

1. Locate the interface front cells, where the sign(φ) is alternating or the value of the volume fraction is between 0 and 1, that is, a partially filled cell.

2. Calculate the normal of the interface segment in each front cell from the level-set function gradients.

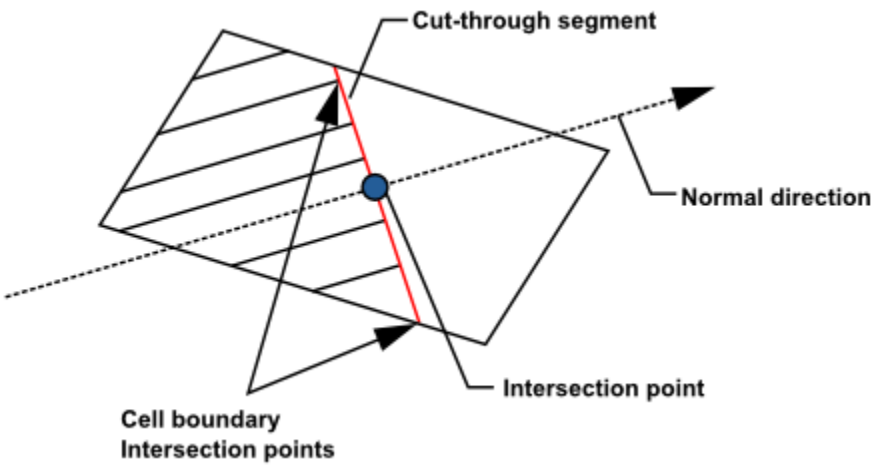
3. Position the cut-through, making sure at least one corner of the cell is occupied by the designated phase in relation to the neighboring cells.

4. Find the intersection between the cell centerline normal to the interface and the interface so that the VOF is satisfied.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

5. Find the intersection points between the interface line and the cell boundaries; these intersection points are designated as front points.

Figure 14.5: Schematic View of the Interface Cut Through the Front Cell



Once the interface front is reconstructed, the procedure for the minimization of the distance from a given point to the interface can begin as follows:

1. Calculate the distance of the given point in the domain to each cut-segment of the front cell. The method for the distance calculation is as follows:

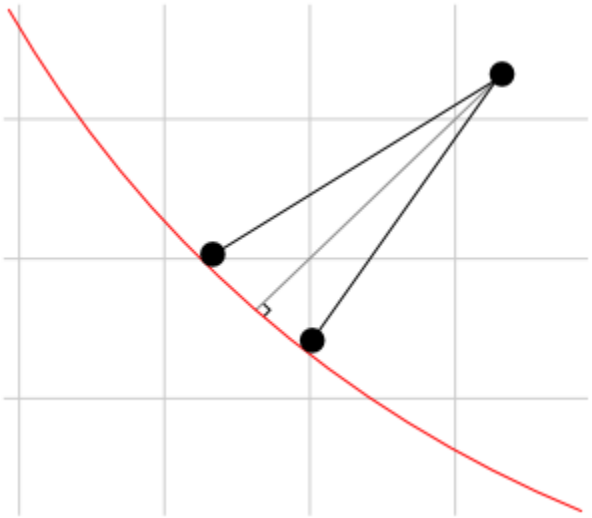
   a. If the normal line starting from the given point to the interface intersects within the cut-segment, then the calculated distance will be taken as the distance to the interface.

   b. If the intersection point is beyond the end points of the cut-segment, the shortest distance from the given point to the end points of the cut-segment will be taken as the distance to the interface cut-segment, as shown in Figure 14.6: Distance to the Interface Segment (p. 640).

2. Minimize all possible distances from the given point to all front cut-segments, so as to represent the distance from the given point to the interface. Thus, the values of these distances will be used to re-initialize the level-set function.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Figure 14.6: Distance to the Interface Segment



14.3.12.2. Limitations

- The current implementation of the model is only suitable for two-phase flow regime, where two fluids are not interpenetrating.
- The level-set model can only be used when the VOF model is enabled.
- The following features are not supported with the coupled VOF Level Set model:
  - Mass transfer
  - Periodic boundaries
  - Polyhedral meshes
  - Dynamic meshes
  - Overset meshes

14.4. Mixture Model Theory

Information is organized into the following subsections:

14.4.1. Overview
14.4.2. Limitations of the Mixture Model
14.4.3. Continuity Equation
14.4.4. Momentum Equation
14.4.5. Energy Equation
14.4.6. Relative (Slip) Velocity and the Drift Velocity
14.4.7. Volume Fraction Equation for the Secondary Phases
14.4.8. Granular Properties
14.4.9. Granular Temperature

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

14.4.10. Solids Pressure
14.4.11. Interfacial Area Concentration
14.4.12. Flow Regime Modeling

14.4.1. Overview

The mixture model is a simplified multiphase model that can be used in different ways. It can be used to model multiphase flows where the phases move at different velocities, but assume local equilibrium over short spatial length scales. It can be used to model homogeneous multiphase flows with very strong coupling and phases moving at the same velocity.

The mixture model can model n phases (fluid or particulate) by solving the momentum, continuity, and energy equations for the mixture, the volume fraction equations for the secondary phases, and algebraic expressions for the relative velocities. Typical applications include sedimentation, cyclone separators, particle-laden flows with low loading, and bubbly flows where the gas volume fraction remains low.

The mixture model is a good substitute for the full Eulerian multiphase model in several cases. A full multiphase model may not be feasible when there is a wide distribution of the particulate phase or when the interphase laws are unknown or their reliability can be questioned. A simpler model like the mixture model can perform as well as a full multiphase model while solving a smaller number of variables than the full multiphase model.

The mixture model allows you to select granular phases and calculates all properties of the granular phases. This is applicable for liquid-solid flows.

14.4.2. Limitations of the Mixture Model

The following limitations apply to the mixture model in Ansys Fluent:

- You must use the pressure-based solver. The mixture model is not available with the density-based solver.
- When the mixture model is used, do not model streamwise periodic flow with specified mass flow rate.
- Do not model solidification and melting in conjunction with the mixture model.
- Do not use the relative formulation in combination with the MRF and mixture model (see Limitations in the User's Guide).
- The mixture model does not allow for inviscid flows.
- When tracking particles with the DPM model in combination with the Mixture model, the **Shared Memory** method cannot be selected (Parallel Processing for the Discrete Phase Model). (Note that using the **Message Passing** or **Hybrid** method enables the compatibility of all multiphase flow models with the DPM model.)

The mixture model, like the VOF model, uses a *single-fluid* approach. It differs from the VOF model in two respects:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

- The mixture model allows the phases to be interpenetrating. The volume fractions α_q and α_p for a control volume can therefore be equal to any value between 0 and 1, depending on the space occupied by phase q and phase p.
- The mixture model allows the phases to move at different velocities, using the concept of slip velocities. (Note that the phases can also be assumed to move at the same velocity, and the mixture model is then reduced to a homogeneous multiphase model.)
- The mixture model is not compatible with non-premixed, partially premixed, and premixed combustion models.

The mixture model solves the continuity equation for the mixture, the momentum equation for the mixture, the energy equation for the mixture, and the volume fraction equation for the secondary phases, as well as algebraic expressions for the relative velocities (if the phases are moving at different velocities).

14.4.3. Continuity Equation

The continuity equation for the mixture is

$$\frac{\partial}{\partial t}\left(\rho_m\right)+\nabla\cdot\left(\rho_m\vec{v}_m\right)=0 \tag{14.119}$$

where $\vec{v}_m$ is the mass-averaged velocity:

$$\vec{v}_m=\frac{\sum_{k=1}^{n}\alpha_k\rho_k\vec{v}_k}{\rho_m} \tag{14.120}$$

and ρ_m is the mixture density:

$$\rho_m=\sum_{k=1}^{n}\alpha_k\rho_k \tag{14.121}$$

α_k is the volume fraction of phase k.

14.4.4. Momentum Equation

The momentum equation for the mixture can be obtained by summing the individual momentum equations for all phases. It can be expressed as

$$\begin{aligned}\frac{\partial}{\partial t}\left(\rho_m\vec{v}_m\right)+\nabla\cdot\left(\rho_m\vec{v}_m\vec{v}_m\right) &= -\nabla p+\nabla\cdot\left[\mu_m\left(\nabla\vec{v}_m+\nabla\vec{v}_m^{T}\right)\right]\\ &\quad+\rho_m\vec{g}+\vec{F}-\nabla\cdot\left(\sum_{k=1}^{n}\alpha_k\rho_k\vec{v}_{dr,k}\vec{v}_{dr,k}\right)\end{aligned} \tag{14.122}$$

where n is the number of phases, $\vec{F}$ is a body force, and μ_m is the viscosity of the mixture:

$$\mu_m=\sum_{k=1}^{n}\alpha_k\mu_k \tag{14.123}$$

$\vec{v}_{dr,k}$ is the drift velocity for secondary phase k:

$$\vec{v}_{dr,k}=\vec{v}_k-\vec{v}_m \tag{14.124}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

14.4.5. Energy Equation

The energy equation for the mixture takes the following form:

$$\frac{\partial}{\partial t}\sum_k\left(\alpha_k\rho_k E_k\right)+\nabla\cdot\sum_k\left(\alpha_k\vec{v}_k\left(\rho_k E_k+p\right)\right)=\nabla\cdot\left(k_{eff}\nabla T-\sum_k\sum_j h_{j,k}\vec{J}_{j,k}+\left(\overline{\overline{\tau}}_{eff}\cdot\vec{v}\right)\right)+S_h \tag{14.125}$$

where $h_{j,k}$ is enthalpy of species j in phase k, $\vec{J}_{j,k}$ is the diffusive flux of species j in phase k, and k_{eff} is the effective conductivity calculated as:

$$k_{eff}=\sum\alpha_k\left(k_k+k_t\right) \tag{14.126}$$

where k_t is the turbulent thermal conductivity defined according to the turbulence model being used.

The first three terms on the right-hand side of Equation 14.125 (p. 643) represent energy transfer due to conduction, species diffusion, and viscous dissipation, respectively. The last term S_h includes volumetric heat sources that you have defined but not the heat sources generated by finite-rate volumetric or surface reactions since species formation enthalpy is already included in the total enthalpy calculation as described in Energy Sources Due to Reaction (p. 174).

In Equation 14.125 (p. 643),

$$E_k=h_k-\frac{p}{\rho_k}+\frac{v^2}{2} \tag{14.127}$$

for a compressible phase, and $E_k=h_k$ for an incompressible phase, where h_k is the sensible enthalpy for phase k.

14.4.6. Relative (Slip) Velocity and the Drift Velocity

The relative velocity (also referred to as the slip velocity) is defined as the velocity of a secondary phase (p) relative to the velocity of the primary phase (q):

$$\vec{v}_{pq}=\vec{v}_p-\vec{v}_q \tag{14.128}$$

The mass fraction for any phase (k) is defined as

$$c_k=\frac{\alpha_k\rho_k}{\rho_m} \tag{14.129}$$

The drift velocity and the relative velocity ($\vec{v}_{qp}$) are connected by the following expression:

$$\vec{v}_{dr,p}=\vec{v}_{pq}-\sum_{k=1}^{n}c_k\vec{v}_{kq} \tag{14.130}$$

where $\vec{v}_{kq}$ is the velocity of phase k relative to phase q.

Ansys Fluent's mixture model makes use of an algebraic slip formulation. The basic assumption of the algebraic slip mixture model is that to prescribe an algebraic relation for the relative velocity, a local equilibrium between the phases should be reached over a short spatial length scale. Following Manninen et al. [408] (p. 1076), the form of the relative velocity is given by:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$\vec{v}_{pq}=\frac{\tau_p}{f_{drag}}\frac{\left(\rho_p-\rho_m\right)}{\rho_p}\vec{a} \tag{14.131}$$

where τ_p is the particle relaxation time

$$\tau_p=\frac{\rho_p d_p^2}{18\mu_q} \tag{14.132}$$

d is the diameter of the particles (or droplets or bubbles) of secondary phase p, $\vec{a}$ is the secondary-phase particle's acceleration. The default drag function f_{drag} is taken from Schiller and Naumann [574] (p. 1086):

$$f_{drag}=\begin{cases}1+0.15Re^{0.687} & Re\leq 1000\\ 0.0183Re & Re>1000\end{cases} \tag{14.133}$$

and the acceleration $\vec{a}$ is of the form

$$\vec{a}=\vec{g}-\left(\vec{v}_m\cdot\nabla\right)\vec{v}_m-\frac{\partial\vec{v}_m}{\partial t} \tag{14.134}$$

The simplest algebraic slip formulation is the so-called drift flux model, in which the acceleration of the particle is given by gravity and/or a centrifugal force and the particulate relaxation time is modified to take into account the presence of other particles.

In turbulent flows the relative velocity should contain a diffusion term due to the dispersion appearing in the momentum equation for the dispersed phase. Ansys Fluent adds this dispersion to the relative velocity:

$$\vec{v}_{pq}=\frac{\left(\rho_p-\rho_m\right)d_p^2}{18\mu_q f_{drag}}\vec{a}-\frac{\eta_t}{\sigma_t}\left(\frac{\nabla\alpha_p}{\alpha_p}-\frac{\nabla\alpha_q}{\alpha_q}\right) \tag{14.135}$$

where σ_t is a Prandtl/Schmidt number set to 0.75 and η_t is the turbulent diffusivity. This diffusivity is calculated from the continuous-dispersed fluctuating velocity correlation, such that

$$\eta_t=C_\mu\frac{k^2}{\epsilon}\left(\frac{\gamma_\gamma}{1+\gamma_\gamma}\right)\left(1+C_\beta\zeta_\gamma^2\right)^{-1/2} \tag{14.136}$$

$$\zeta_\gamma=\frac{\left|\vec{v}_{pq}\right|}{\sqrt{2/3k}} \tag{14.137}$$

where γ_γ is the time ratio between the time scale of the energetic turbulent eddies affected by the crossing-trajectories effect and the particle relaxation time, $C_\beta=1.8-1.35\cos^2\theta$, and $\cos\theta=\left(\vec{v}_{pq}\cdot\vec{v}_p\right)/\left(\left|\vec{v}_{pq}\right|\left|\vec{v}_p\right|\right)$.

When you are solving a mixture multiphase calculation with slip velocity, you can directly prescribe formulations for the drag function. The following choices are available:

- Schiller-Naumann (the default formulation)
- Morsi-Alexander

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

- symmetric
- Grace et al.
- Tomiyama et al.
- universal drag laws
- constant
- user-defined

See Interphase Exchange Coefficients (p. 667) for more information on these drag functions and their formulations, and Defining the Phases for the Mixture Model in the User's Guide for instructions on how to enable them.

Note that, if the slip velocity is not solved, the mixture model is reduced to a homogeneous multiphase model. In addition, the mixture model can be customized (using user-defined functions) to use a formulation other than the algebraic slip method for the slip velocity. See the Fluent Customization Manual for details.

14.4.7. Volume Fraction Equation for the Secondary Phases

From the continuity equation for secondary phase p, the volume fraction equation for secondary phase p can be obtained:

$$\frac{\partial}{\partial t}\left(\alpha_p\rho_p\right)+\nabla\cdot\left(\alpha_p\rho_p\vec{v}_m\right)=-\nabla\cdot\left(\alpha_p\rho_p\vec{v}_{dr,p}\right)+\sum_{q=1}^{n}\left(\dot{m}_{qp}-\dot{m}_{pq}\right) \tag{14.138}$$

14.4.8. Granular Properties

Since the concentration of particles is an important factor in the calculation of the effective viscosity for the mixture, we may use the granular viscosity to get a value for the viscosity of the suspension. The volume weighted averaged for the viscosity would now contain shear viscosity arising from particle momentum exchange due to translation and collision.

The collisional and kinetic parts, and the optional frictional part, are added to give the solids shear viscosity:

$$\mu_s=\mu_{s,col}+\mu_{s,kin}+\mu_{s,fr} \tag{14.139}$$

Refer to Solids Pressure (p. 694) for definitions of the constants used in granular flow.

14.4.8.1. Collisional Viscosity

The collisional part of the shear viscosity is modeled as [205] (p. 1064), [637] (p. 1090).

$$\mu_{s,col}=\frac{4}{5}\alpha_s\rho_s d_s g_{0,ss}\left(1+e_{ss}\right)\left(\frac{\Theta_s}{\pi}\right)^{1/2} \tag{14.140}$$

14.4.8.2. Kinetic Viscosity

Ansys Fluent provides two expressions for the kinetic viscosity.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

The default expression is from Syamlal et al. [637] (p. 1090):

$$\mu_{s,kin}=\frac{d_s\rho_s\sqrt{\Theta_s\pi}}{6(3-e_{ss})}\left[1+\frac{2}{5}(1+e_{ss})(3e_{ss}-1)\alpha_s g_{0,ss}\right] \tag{14.141}$$

The following optional expression from Gidaspow et al. [205] (p. 1064) is also available:

$$\mu_{s,kin}=\frac{10\rho_s d_s\sqrt{\Theta_s\pi}}{96\alpha_s(1+e_{ss})g_{0,ss}}\left[1+\frac{4}{5}g_{0,ss}\alpha_s(1+e_{ss})\right]^2 \tag{14.142}$$

14.4.9. Granular Temperature

The viscosities need the specification of the granular temperature for the s^{th} solids phase. Here we use an algebraic equation from the **granular temperature** transport equation. This is only applicable for dense fluidized beds where the convection and the diffusion term can be neglected under the premise that production and dissipation of granular energy are in equilibrium.

$$0=\left(-p_s\overline{\overline{I}}+\overline{\overline{\tau}}_s\right):\nabla\vec{v}_s-\gamma_{\Theta_s}+\varphi_{ls} \tag{14.143}$$

where

$\left(-p_s\overline{\overline{I}}+\overline{\overline{\tau}}_s\right):\nabla\vec{v}_s$ = the generation of energy by the solid stress tensor

γ_{Θ_s} = the collisional dissipation of energy

φ_{ls} = the energy exchange between the l^{th} fluid or solid phase and the s^{th} solid phase

The collisional dissipation of energy, γ_{Θ_s}, represents the rate of energy dissipation within the s^{th} solids phase due to collisions between particles. This term is represented by the expression derived by Lun et al. [394] (p. 1075)

$$\gamma_{\Theta m}=\frac{12(1-e_{ss}^2)g_{0,ss}}{d_s\sqrt{\pi}}\rho_s\alpha_s^2\Theta_s^{3/2} \tag{14.144}$$

The transfer of the kinetic energy of random fluctuations in particle velocity from the s^{th} solids phase to the l^{th} fluid or solid phase is represented by φ_{ls} [205] (p. 1064):

$$\varphi_{ls}=-3K_{ls}\Theta_s \tag{14.145}$$

Ansys Fluent allows you to solve for the granular temperature with the following options:

- algebraic formulation (the default)

  This is obtained by neglecting convection and diffusion in the transport equation (Equation 14.143 (p. 646)) [637] (p. 1090).

- constant granular temperature

  This is useful in very dense situations where the random fluctuations are small.

- UDF for granular temperature

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

14.4.10. Solids Pressure

The total non-filtered solid pressure is calculated and included in the mixture momentum equations:

$$P_{s,total} = \sum_{q=1}^{N} p_q \tag{14.146}$$

where p_q is presented in the section for granular flows by Equation 14.349 (p. 695).

14.4.11. Interfacial Area Concentration

Interfacial area concentration is defined as the interfacial area between two phases per unit mixture volume. This is an important parameter for predicting mass, momentum and energy transfers through the interface between the phases. When using the Mixture multiphase model with non-granular secondary phases, you can have Ansys Fluent compute the interfacial area in one of following ways:

- use a transport equation for interfacial area concentration as further described in Transport Equation Based Models (p. 647). This allows for a distribution of bubble diameters and coalescence/breakage effects.
- use an algebraic relationship between a specified bubble diameter and the interfacial area density. For more information, see Algebraic Models (p. 651).

The key difference between the transport equation based interfacial area concentration (IAC) models (Transport Equation Based Models (p. 647)) and algebraic models (Algebraic Models (p. 651)) is that the algebraic models assume the interface to be spherical, whereas the IAC models can predict the interface area concentration directly through the solution of a transport equation

14.4.11.1. Transport Equation Based Models

In two-fluid flow systems, one discrete (particles) and one continuous, the size and its distribution of the discrete phase or particles can change rapidly due to growth (mass transfer between phases), expansion due to pressure changes, coalescence, breakage and/or nucleation mechanisms. The Population Balance model (see Population Balance Model (p. 785)) ideally captures this phenomenon, but is computationally expensive since several transport equations need to be solved using moment methods, or more if the discrete method is used. The interfacial area concentration model uses a single transport equation per secondary phase and is specific to bubbly flows in liquid at this stage.

The transport equation for the interfacial area concentration can be written as

$$\frac{\partial\left(\rho_g \chi_p\right)}{\partial t} + \nabla\cdot\left(\rho_g \vec{u}_g \chi_p\right) = \frac{1}{3}\frac{D\rho_g}{Dt}\chi_p + \frac{2}{3}\frac{\dot{m}_g}{\alpha_g}\chi_p + \rho_g\left(S_{RC}+S_{WE}+S_{TI}\right) \tag{14.147}$$

where χ_p is the interfacial area concentration (m^2/m^3), and α_g is the gas volume fraction. The first two terms on the right hand side of Equation 14.147 (p. 647) are of gas bubble expansion due to compressibility and mass transfer (phase change). $\dot{m}_g$ is the mass transfer rate into the gas phase per unit mixture volume ($kg/m^3/s$). S_{RC} and S_{WE} are the coalescence sink terms due to random collision and wake entrainment, respectively. S_{TI} is the breakage source term due to turbulent impact.

Two sets of models, the Hibiki-Ishii model [243] (p. 1067) and the Ishii-Kim model [304] (p. 1070), [271] (p. 1068), exist for those source and sink terms for the interfacial area concentration, which are

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

based on the works of Ishii et al. [243] (p. 1067), [304] (p. 1070). According to their study, the mechanisms of interactions can be summarized in five categories:

- Coalescence due to random collision driven by turbulence.
- Breakage due to the impact of turbulent eddies.
- Coalescence due to wake entrainment.
- Shearing-off of small bubbles from large cap bubbles.
- Breakage of large cap bubbles due to flow instability on the bubble surface.

In Ansys Fluent, only the first three effects will be considered.

14.4.11.1.1. Hibiki-Ishii Model

$$\begin{aligned} S_{RC} &= -\frac{1}{3\varphi}\left(\frac{\alpha_g}{\chi_p}\right)^2 f_c n_b \lambda_c \\ &= -\left(\frac{\alpha_g}{\chi_p}\right)^2 \frac{\Gamma_c \alpha_g^2 \epsilon^{1/3}}{d_b^{11/3}\left(\alpha_{gmax}-\alpha_g\right)} exp\left(-K_c \frac{d_b^{5/6}\rho_f^{1/2}\epsilon^{1/3}}{\sigma^{1/2}}\right) \\ &= -\frac{\Gamma_c}{\psi^{11/3}} \frac{\epsilon^{1/3}}{\left(\alpha_{gmax}-\alpha_g\right)} \alpha_g^{1/3} \chi_p^{5/3} exp\left[-K_c \psi^{5/6} \frac{\rho_f^{1/2}\epsilon^{1/3}}{\sigma^{1/2}} \left(\frac{\alpha_g}{\chi_p}\right)^{5/6}\right] \end{aligned} \tag{14.148}$$

where f_c, λ_c and n_b are the frequency of particle/bubble collision, the efficiency of coalescence from the collision, and the number of particles per unit mixture volume, respectively. The averaged size of the particle/bubble d_b is assumed to be calculated as:

$$d_b = \psi \frac{\alpha_g}{\chi_p} \tag{14.149}$$

and

$$\lambda_c = exp\left(-K_c \frac{d_b^{5/6}\rho_f^{1/2}\varepsilon^{1/3}}{\sigma^{1/2}}\right) \tag{14.150}$$

$$\begin{aligned} S_{TI} &= \frac{1}{3\varphi}\left(\frac{\alpha_g}{\chi_p}\right)^2 f_B n_e \lambda_B \\ &= \left(\frac{\alpha_g}{\chi_p}\right)^2 \frac{\Gamma_B \alpha_g \left(1-\alpha_g\right)\epsilon^{1/3}}{d_b^{11/3}\left(\alpha_{gmax}-\alpha_g\right)} exp\left(-K_B \frac{\sigma}{\rho_f d_b^{5/3}\epsilon^{2/3}}\right) \\ &= \frac{\Gamma_B}{\psi^{11/3}} \frac{\left(1-\alpha_g\right)\epsilon^{1/3}\chi_p^{5/3}}{\alpha_g^{2/3}\left(\alpha_{gmax}-\alpha_g\right)} exp\left[-\frac{K_B}{\psi^{5/3}} \frac{\sigma}{\rho_f \epsilon^{2/3}} \left(\frac{\chi_p}{\alpha_g}\right)^{5/3}\right] \end{aligned} \tag{14.151}$$

where f_B, λ_B and n_e are the frequency of collision between particles/bubbles and turbulent eddies of the primary phase, the efficiency of breakage from the impact, and the number of turbulent eddies per unit mixture volume, respectively. In Equation 14.151 (p. 648),

$$\lambda_B = exp\left(-K_B \frac{\sigma}{\rho_f d_b^{5/3}\varepsilon^{2/3}}\right) \tag{14.152}$$

The experimental adjustable coefficients are given as follows:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Γ_C=0.188; K_C=0.129; Γ_B=0.264; K_B=1.37.

Note that based on experimental data, Ansys Fluent uses a different value for K_C than that proposed by Hibiki and Ishii [243] (p. 1067). If you want to modify this value for your case, contact Ansys Technical Support for information on how to customize your settings.

The shape factor ψ is given as 6 and φ as $\frac{1}{36\pi}$ for spherical particles/bubbles. There is no model for S_{WE} in the Hibiki-Ishii formulation.

14.4.11.1.2. Ishii-Kim Model

$$\begin{aligned} S_{RC} &= -\frac{1}{3\varphi}\left(\frac{\alpha_g}{\chi_p}\right)^2 C_{RC}\left[\frac{n_b^2 u_t d_b^2}{\alpha_{gmax}^{1/3}\left(\alpha_{gmax}^{1/3}-\alpha_g^{1/3}\right)}\right]\left[1-exp\left(-C\frac{\alpha_{gmax}^{1/3}\alpha_g^{1/3}}{\alpha_{gmax}^{1/3}-\alpha_g^{1/3}}\right)\right] \\ &= -\frac{1}{3\pi} C_{RC} u_t \chi_p^2\left[\frac{1}{\alpha_{gmax}^{1/3}\left(\alpha_{gmax}^{1/3}-\alpha_g^{1/3}\right)}\right]\left[1-exp\left(-C\frac{\alpha_{gmax}^{1/3}\alpha_g^{1/3}}{\alpha_{gmax}^{1/3}-\alpha_g^{1/3}}\right)\right] \end{aligned} \tag{14.153}$$

$$S_{WE} = -\frac{1}{3\varphi}\left(\frac{\alpha_g}{\chi_p}\right)^2 n_b^2 d_b^2 u_r C_D^{1/3} = -\frac{1}{3\pi} C_{WE} u_r \chi_p^2 C_D^{1/3} \tag{14.154}$$

$$\begin{aligned} S_{TI} &= \frac{1}{3\varphi}\left(\frac{\alpha_g}{\chi_p}\right)^2 C_{TI}\left(\frac{n_b u_t}{d_b}\right)\left(1-\frac{We_{cr}}{We}\right)^{1/2} exp\left(-\frac{We_{cr}}{We}\right) \\ &= \frac{1}{18} C_{TI} u_t \frac{\chi_p^2}{\alpha_g}\left(1-\frac{We_{cr}}{We}\right)^{1/2} exp\left(-\frac{We_{cr}}{We}\right) \end{aligned} \tag{14.155}$$

where the mean bubble fluctuating velocity, u_t, is given by $\varepsilon^{1/3} d_b^{1/3}$. The bubble terminal velocity, u_r, is a function of the bubble diameter and local time-averaged void fraction.

$$u_r = \left(\frac{d_b g \Delta\rho}{3 C_D \rho_f}\right)^{1/2} \tag{14.156}$$

$$C_D = 24\frac{\left(1+0.1Re_D^{0.75}\right)}{Re_D} \quad and \quad Re_D \equiv \frac{\rho_f u_r d_b}{\mu_f}\left(1-\alpha_g\right) \tag{14.157}$$

$$We = \frac{\rho_f u_t^2 d_b}{\sigma} \tag{14.158}$$

where μ_f is the molecular viscosity of the fluid phase, g is the gravitational acceleration and σ is the interfacial tension. In this model, when the Weber number, We, is less than the critical Weber number, We_{cr}, the breakage rate equals zero, that is, S_{TI}=0. The coefficients used are given as follows [271] (p. 1068):

C_{RC} = 0.004
C_{WE} = 0.002
C_{TI} = 0.085
C = 3.0
We_{cr} = 6.0

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$\alpha_{g\max} = 0.75$

Important:

Currently, this model is only suitable for two-phase flow regimes, one phase being gas and another liquid, that is, bubbly column applications. However, you can always use UDFs to include your own interfacial area concentration models, which can apply to other flow regimes.

See the Fluent Customization Manual for details.

14.4.11.1.3. Yao-Morel Model

The volumetric interfacial area is an important quantity that appears in the calculation of the interphase exchange forces like momentum, mass, and heat transfer. In Ansys Fluent, the Hibiki and Ishii model ([243] (p. 1067)) and the Ishii and Kim model (Hibiki-Ishii Model (p. 648)) have been implemented In the context of bubbly flows with bulk mass transfer. An extension to these two models exists. This will include heterogeneous mass transfer effects on wall based on the work done by Yao and Morel [719] (p. 1094) for nucleate boiling applications. The volumetric interfacial area transport Equation 14.147 (p. 647) contains a nucleation term and models for coalescence and breakup. Yao and Morel [719] (p. 1094) modeled the coalescence term as follows:

$$\Phi^{CO} = \frac{1}{2}\frac{\eta_c n}{T_{cf} + T_{ci}} \tag{14.159}$$

where T_{cf} and T_{cI} are the free traveling time and the interaction time for coalescence, n is the bubble number density, η_c is the coalescence efficiency and the factor one-half has been included to avoid double counting on the same events between bubble pairs. The final expression for the above equation is

$$\Phi^{CO} = -K_{c1}\frac{\varepsilon^{1/3}\alpha^2}{d_b^{11/3}}\frac{1}{g(\alpha) + K_{c2}\alpha\sqrt{\frac{We}{We_c}}}\cdot\exp\left(K_{c3}\sqrt{\frac{We}{We_c}}\right) \tag{14.160}$$

where $We_c = 1.24$ is the critical Weber number and the coefficients have the values $K_{c1} = 2.86$, $K_{c2} = 1.922$, $K_{c3} = 1.017$, We is the Weber number, ε is the dissipation obtained from the k-epsilon turbulence model, $g(\alpha)$ is a modification factor defined as $1 - \left({}^{\alpha}/_{\alpha_{\max}}\right)^{1/3}$ and $\alpha_{\max} = 0.52$ is the packing limit. Regarding Equation 14.147 (p. 647), the coalescence sink terms due to random collision and wake entrainment are as follows:

$$S_{RC} + S_{WE} = \frac{1}{3\varphi}\left(\frac{\alpha}{\chi_p}\right)^2 \Phi^{CO} \tag{14.161}$$

The breakage term, as modeled by Yao and Morel [719] (p. 1094), is

$$\Phi^{BK} = \frac{\eta_b n}{T_{bf} + T_{bi}} \tag{14.162}$$

where T_{bf} and T_{bi} are the free traveling time and the interaction time for breakage and η_b is the breakage efficiency. The final expression for the above equation is of the form

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$\Phi^{BK}=K_{b1}\frac{\varepsilon^{1/3}\alpha\left(1-\alpha\right)}{d_b^{11/3}}\frac{1}{1+K_{b2}\left(1-\alpha\right)\sqrt{\frac{We}{We_c}}}\cdot\exp\left(-\frac{We}{We_c}\right) \tag{14.163}$$

Here, the coefficients have the values $K_{b1}=1.6$ and $K_{b2}=0.42$

Regarding Equation 14.147 (p. 647), the S_{TI} breakage source term due to turbulent impact is of the form

$$S_{TI}=\frac{1}{3\varphi}\left(\frac{\alpha}{\chi_p}\right)^2\Phi^{BK} \tag{14.164}$$

When including source terms due to bubble nucleation at the heated wall a new term should be added to Equation 14.147 (p. 647). This is of the form

$$\Phi^{NUC}=\pi d_{bw}^2 N_w f\frac{A_{heated}}{Volume} \tag{14.165}$$

where d_{bw} is the diameter of the nucleation bubble, N_w is the nucleation site density, and f is the frequency of the bubble. These parameters can be connected to the boiling models described in Wall Boiling Models (p. 733). The bubble departure frequency is given by Equation 14.514 (p. 735), the nucleate site density by Equation 14.515 (p. 735), and the bubble departure diameter by Equation 14.518 (p. 736), Equation 14.519 (p. 736), or Equation 14.520 (p. 736).

To learn how to apply this model, refer to Defining the Interfacial Area Concentration via the Transport Equation in the User's Guide.

14.4.11.2. Algebraic Models

The algebraic models are derived from the surface area to volume ratio, A_p, for a spherical bubble or droplet:

$$A_p=\frac{\pi d_p^2}{\frac{1}{6}\pi d_p^3}=\frac{6}{d_p} \tag{14.166}$$

where d_p is the bubble or droplet diameter. Although the algebraic models are derived based on the assumption that the bubble or droplet has a spherical shape, the unit of the interfacial area density A_p is the same as that of χ_p in the transported equation based models. When using the Mixture multiphase model with non-granular secondary phases, you can have Ansys Fluent compute the interfacial area in one of the following ways:

- Symmetric model (default)

  The symmetric model treats both phases p and q symmetrically. Phases p and q can be continuous or dispersed. The interphase area density is calculated as:

  $$A_i=\frac{6\alpha_q\alpha_p}{d_{p,q}} \tag{14.167}$$

  where α_p and α_q are the volume fraction of the phase p and phase q, respectively, and $d_{p,q}$ is the characteristic length scale computed as follows:

  – If the phase p is a dispersed phase:

$$d_{p,q}=d_p \tag{14.168}$$

– If both phases p and q are dispersed:

$$d_{p,q}=\alpha_q d_p+\alpha_p d_q \tag{14.169}$$

- Particle model

  For a dispersed phase, p , with volume fraction, α_p , the particle model estimates the interfacial area density, A_i as

$$A_i=\alpha_p A_p=\frac{6\alpha_p}{d_p} \tag{14.170}$$

- Gradient model

  The Gradient model differs from the Symmetric model in that it introduces the gradient at the phase interface as the interfacial length scale. This model is typically used for free surface problems. Two versions of the Gradient model are available in Ansys Fluent:

  – Pure gradient model

    If there is a free surface between the continuous phase p and dispersed phase q, then the interfacial area density is computed as:

$$A_i=\left|\nabla \alpha_q\right| \tag{14.171}$$

    For dispersed-dispersed systems with more than two phases, the interfacial area density for phases p and q is expressed as:

$$A_i=\frac{2\left|\nabla \alpha_q\right|\cdot\left|\nabla \alpha_p\right|}{\left|\nabla \alpha_q\right|+\left|\nabla \alpha_p\right|} \tag{14.172}$$

  – Gradient-Symmetric model

    The Gradient-Symmetric model computes the interfacial area density based on the free surface position as follows. If $\left|\nabla \alpha_q\right| / \max\left|\nabla \alpha_q\right|$ < specified limit (default 1e-6), then the interfacial area density is computed using the Symmetric model Equation 14.167 (p. 651). Otherwise, the interfacial area density is set to zero.

- User-Defined

  See `DEFINE_EXCHANGE_PROPERTY` in the Fluent Customization Manual.

14.4.12. Flow Regime Modeling

While the algebraic slip model supports only small-scale interfaces in the low Stokes number regime, the generalized flow regime framework effectively handles both the small and large-scale interfaces along with the regime transition using special numerical methods. This is important for accurate simulations of multiphase applications that involve various flow regimes, such as free surface flow, bubbly flow, and droplet flow. For more information on how to use the flow regime modeling, refer to Using the Flow Regime Modeling in the *Fluent User's Guide*.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

The generalized flow regime framework distinguishes three different flow regimes (that is, bubbly flow, droplet flow, and free surface flow) and has the following salient features:

- Definition of the flow regime by the phase morphology and phase state.
- Hybrid morphology specification based on the flow regime detection criterion.
- Support of multiple continuous and dispersed phases.
- Different sets of methods and defaults for different phase-pair morphologies.
- Regime detection via the Algebraic Interfacial Area Density (AIAD) method along with smoothing and filtering techniques.
- Ability to use user-defined functions for flow regime modeling.
- Regime-based volume fraction discretization.

In the flow regime modeling, phase morphology, which represents the connectivity of the fluid, is classified in the following categories:

- Continuous

  A phase morphology is classified as continuous if the phase occupies a connected region in the flow domain.

- Dispersed

  A phase morphology is classified as disperse if the phase occupies disconnected regions in the flow domain.

- Hybrid

  Hybrid morphology is a dynamic morphology that can behave either as continuous or dispersed depending on the flow regime detection criterion.

  Hybrid phase morphology can be observed when:

  – Phase morphology varies in space and/or time, but there is no real flow regime transition.

  – There is a regime transition between bubbly flow, droplet flow, and free surface flow.

  Figure 14.7: Hybrid Phase Morphology with no Regime Transition (p. 654) shows an example of a two-phase flow where air has a hybrid morphology, and water has a continuous morphology. No regime transition occurs in the fluid domain.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Figure 14.7: Hybrid Phase Morphology with no Regime Transition

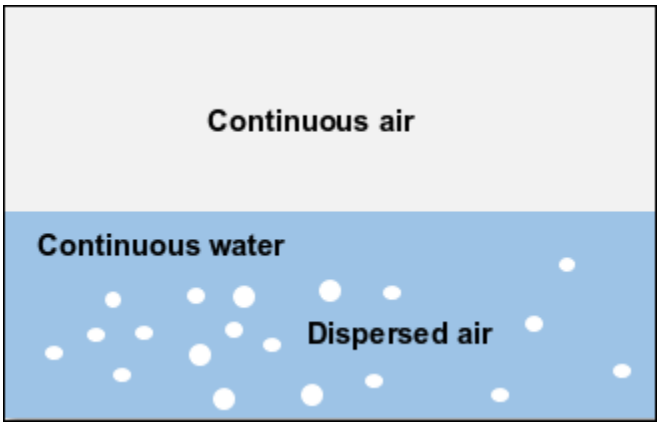
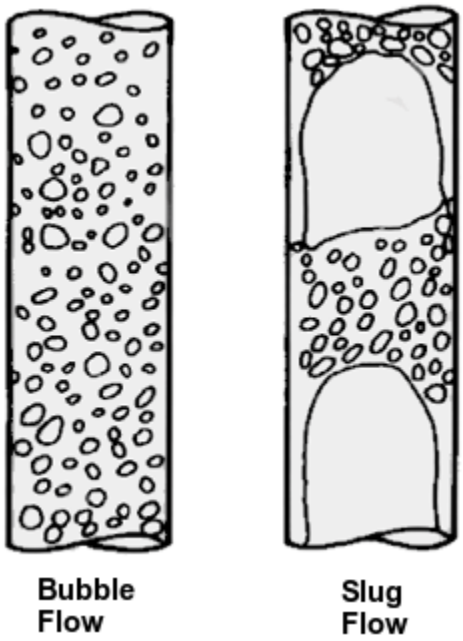


Figure 14.8: Hybrid Phase Morphology with Regime Transition (p. 654) shows an example of a regime transition in a two-phase flow where air behaves as a hybrid phase, while water can behave as a continuous or hybrid phase. The flow regime may transition between bubbly and slug flows.

Figure 14.8: Hybrid Phase Morphology with Regime Transition



Both the phase state (liquid or gas) and phase morphology (continuous or dispersed) are critical for the complete definition of the flow regime, such as free surface flow, bubbly, and droplet flow.

Table 14.2: Flow Regime Definitions (p. 654) shows possible regime definitions based on phase morphology and phase states for different flows.

Table 14.2: Flow Regime Definitions

Flow Regime Classification	Phase Morphology + Phase State
Free Surface Flow	Continuous Liquid/Continuous Liquid

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Flow Regime Classification	Phase Morphology + Phase State
	Continuous Liquid/Continuous Gas
Bubbly Flow	Continuous Liquid/Dispersed Gas
Droplet Flow	Continuous Gas/Dispersed Liquid Continuous Liquid/Dispersed Liquid

14.4.12.1. Phase Pair Interaction for Hybrid Morphology

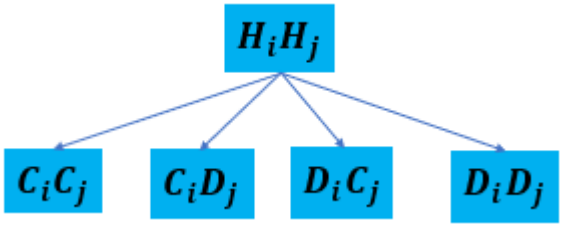
For a pair of phases with phase indices i and j, the following phase-pair interactions can exist:

- Hybrid-Hybrid
- Continuous-Hybrid
- Hybrid-Dispersed

Details for each phase pair interaction are presented below.

Hybrid-Hybrid Phase Pair Interaction

Interactions possible for the Hybrid-Hybrid phase pair are shown in the diagram below.



Here, C = continuous regime, D = dispersed regime, H = hybrid regime.

In the above diagram,

C_iC_j is the interaction where both phases behave as continuous (as in the case of free surface flow). This interaction will be further referred to as CC.

C_iD_j is the interaction where phase i behaves as continuous, and phase j behaves as dispersed (as in the case of bubbly or droplet flow, depending on the phase states). This interaction will be further referred to as CD.

D_iC_j is the interaction where phase i behaves as dispersed and phase j behaves as continuous (as in the case of droplet or bubbly flow, depending on the phase states). This interaction will be further referred to as DC.

D_iD_j is the interaction where both phases behave as dispersed (as in the case of a particle-particle interaction, which is only possibly only for flows with more than two phases). This interaction will be further referred to as DD.

Depending on the flow regime detection criterion, blending factors are calculated to identify various flow regimes that satisfy the following constraint:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$F_{CC} + F_{CD} + F_{DC} + F_{DD} = 1 \tag{14.173}$$

where F is the blending factor, and the subscripts represent various flow regimes as described above.

Area density can be then blended as:

$$A_{ij} = F_{CC}A_{CC} + F_{CD}A_{CD} + F_{DC}A_{DC} + F_{DD}A_{DD} \tag{14.174}$$

where A_{CC} is the area density for free surface flow, A_{CD} and A_{DC} are the area densities for bubbly flow and droplet flow, respectively, and A_{DD} is the area density for particle-particle flow.

For the algebraic slip velocity model with a hybrid-hybrid phase pair, a general form of the slip velocity is given as:

$$\overrightarrow{V_{ij}} = F_{C_iC_j}\overrightarrow{V_{C_iC_j}} + F_{C_iD_j}\overrightarrow{V_{C_iD_j}} + F_{D_iC_j}\overrightarrow{V_{D_iC_j}} + F_{D_iD_j}\overrightarrow{V_{D_iD_j}} \tag{14.175}$$

For Continuous-Continuous interaction, velocities of phases i and j are forced to be equal ($\overrightarrow{V_{C_j}} = \overrightarrow{V_{C_i}}$) by providing zero resistance to the drag, which corresponds to an infinite drag.

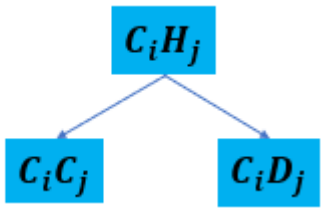
For Dispersed-Dispersed interaction,

$$\overrightarrow{V_{D_iD_j}} = \overrightarrow{V_{D_iC_j}} + \overrightarrow{V_{C_iD_j}} \tag{14.176}$$

For Continuous-Dispersed or Dispersed-Continuous interaction, slip velocity is specified by Equation 14.131 (p. 644) and Equation 14.135 (p. 644).

Continuous-Hybrid Phase Pair Interaction

Interactions possible for the Continuous-Hybrid phase pair where phase i is continuous, and phase j is hybrid are shown in the diagram below.

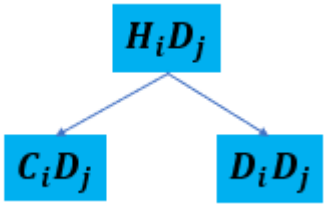


In this case, F_{DC}=0, and F_{DD} = 0.

Hybrid-Dispersed Phase Pair Interaction

Interactions possible for the Continuous-Dispersed phase pair where phase i is hybrid, and phase j is dispersed are shown in the diagram below.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.



In this case, $F_{DC}=0$ and $F_{CC}=0$.

14.4.12.2. Algebraic Interfacial Area Density Method for Flow Regime Blending

Ansys Fluent uses the customized variant of the algebraic interfacial area density method proposed by Thomas Höhne et al. [253] (p. 1067) for the flow regime detection. In this method, various cosine functions are used to calculate blending factors for flow regime detection.

For a pair of phases i and j, the weighting functions f_i and f_j are based on the volume fractions and defined as:

$$f_i = \frac{1}{2}\left[1+cos\left(\pi \frac{\widetilde{\alpha_i}-(\alpha_{i,crit}-\delta_\alpha)}{2\delta_\alpha}\right)\right]$$
$$f_j = \frac{1}{2}\left[1+cos\left(\pi \frac{\widetilde{\alpha_j}-(\alpha_{j,crit}-\delta_\alpha)}{2\delta_\alpha}\right)\right] \quad (14.177)$$

where:

$$\widetilde{\alpha_i} = min\left(max\left(\frac{\alpha_i}{\alpha_i+\alpha_j}, \left(\alpha_{i,crit}-\delta_\alpha\right)\right), \left(\alpha_{i,crit}+\delta_\alpha\right)\right)$$
$$\widetilde{\alpha_j} = min\left(max\left(\frac{\alpha_j}{\alpha_i+\alpha_j}, \left(\alpha_{j,crit}-\delta_\alpha\right)\right), \left(\alpha_{j,crit}+\delta_\alpha\right)\right) \quad (14.178)$$

In the above equations:

α_i and α_j = volume fractions of phase i and phase j, respectively

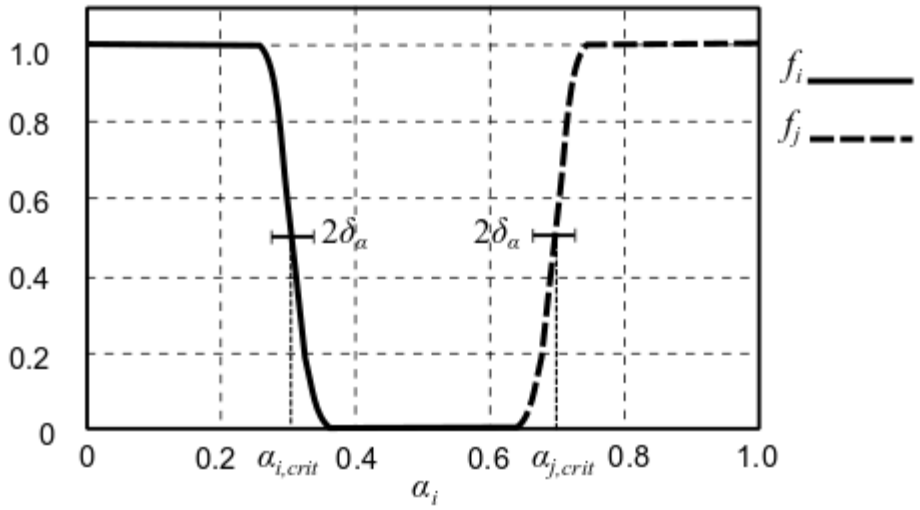
$\alpha_{i,crit}$ and $\alpha_{j,crit}$ = critical volume fractions of phase i and phase j, respectively. Default: 0.3

δ_α = volume fraction width for the intermediate transition. Default: 0.05.

Figure 14.9: Weighting Functions f_i and f_j (p. 658) (adapted from [253] (p. 1067)) shows a plot of the weighting functions f_i and f_j against the volume fraction.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Figure 14.9: Weighting Functions f_i and f_j



The weighting function of the phase boundary f_s based on the volume fraction gradient is expressed as:

$$f_s = \frac{1}{2}\left[1+cos\left(\pi\frac{\widetilde{\nabla\alpha}-\left(\nabla\alpha_{crit}-\delta_\nabla\right)}{2\delta_\nabla}\right)\right] \tag{14.179}$$

where:

$$\widetilde{\nabla\alpha} = min\left(max\left(\left|\nabla\alpha\right|,\ \left(\nabla\alpha_{crit}-\delta_\nabla\right)\right),\left(\nabla\alpha_{crit}+\delta_\nabla\right)\right) \tag{14.180}$$

$$\left|\nabla\alpha\right| = \frac{\left|\nabla\alpha_i\right|\left|\nabla\alpha_j\right|}{\left|\nabla\alpha_i\right|+\left|\nabla\alpha_j\right|} \tag{14.181}$$

$$\nabla\alpha_{crit} = \frac{1}{n\Delta s} \tag{14.182}$$

$$\delta_\nabla = \beta\nabla\alpha_{crit} \tag{14.183}$$

In Equation 14.183 (p. 658), β is the fraction of the critical gradient for defining the intermediate transition width (default: 0.1).

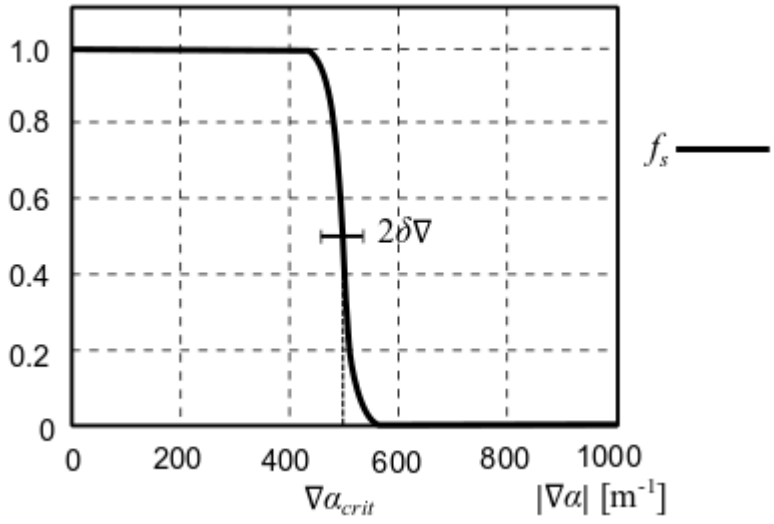
In Equation 14.182 (p. 658), n is the cell number for determining the interfacial width for free surface flow, and Δs is the cell size expressed as:

$$\Delta s = \begin{cases} \sqrt[2]{V} & for\ 2D \\ \sqrt[3]{V} & for\ 3D \end{cases} \tag{14.184}$$

Figure 14.10: Weighting Function f_s (p. 659) (adapted from [253] (p. 1067)) shows the weighting function f_s over the critical gradient of the phase fraction $\nabla\alpha_{crit}$.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Figure 14.10: Weighting Function f_s



For the detection of a phase boundary by the criteria of the volume fraction and its gradient, the weighting function of the phase boundary ψ is used:

$$\psi = f_s\left(f_j - f_i\right) \tag{14.185}$$

The flow regime blending factors are then calculated as:

$$F_{CC} = \left(1 - |\psi|\right) \tag{14.186}$$

$$F_{CD} = \left(1 - a_{sign}\right)|\psi| \tag{14.187}$$

$$F_{DC} = a_{sign}|\psi| \tag{14.188}$$

where

$$a_{sign} = \begin{cases} 1, & if \ \ sign(\psi) = 1 \\ 0, & otherwise \end{cases} \tag{14.189}$$

14.5. Eulerian Model Theory

Details about the Eulerian multiphase model are presented in the following subsections:

14.5.1. Overview of the Eulerian Model
14.5.2. Limitations of the Eulerian Model
14.5.3. Volume Fraction Equation
14.5.4. Conservation Equations
14.5.5. Surface Tension and Adhesion for the Eulerian Multiphase Model
14.5.6. Interfacial Area Concentration
14.5.7. Interphase Exchange Coefficients
14.5.8. Lift Coefficient Correction
14.5.9. Lift Force
14.5.10. Wall Lubrication Force

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

14.5.11. Turbulent Dispersion Force
14.5.12. Virtual Mass Force
14.5.13. Solids Pressure
14.5.14. Maximum Packing Limit in Binary Mixtures
14.5.15. Solids Shear Stresses
14.5.16. Granular Temperature
14.5.17. Description of Heat Transfer
14.5.18. Turbulence Models
14.5.19. Solution Method in Ansys Fluent
14.5.20. Algebraic Interfacial Area Density (AIAD) Model
14.5.21. Generalized Two Phase (GENTOP) Flow Model
14.5.22. The Filtered Two-Fluid Model
14.5.23. Dense Discrete Phase Model
14.5.24. Multi-Fluid VOF Model
14.5.25. Wall Boiling Models

14.5.1. Overview of the Eulerian Model

The Eulerian multiphase model in Ansys Fluent allows for the modeling of multiple separate, yet interacting phases. The phases can be liquids, gases, or solids in nearly any combination. An Eulerian treatment is used for each phase, in contrast to the Eulerian-Lagrangian treatment that is used for the discrete phase model.

With the Eulerian multiphase model, the number of secondary phases is limited only by memory requirements and convergence behavior. Any number of secondary phases can be modeled, provided that sufficient memory is available. For complex multiphase flows, however, you may find that your solution is limited by convergence behavior. See Eulerian Model in the User's Guide for multiphase modeling strategies.

Ansys Fluent's Eulerian multiphase model does not distinguish between fluid-fluid and fluid-solid (granular) multiphase flows. A granular flow is simply one that involves at least one phase that has been designated as a granular phase.

The Ansys Fluent solution is based on the following:

- A single pressure is shared by all phases.
- Momentum and continuity equations are solved for each phase.
- The following parameters are available for granular phases:
  - Granular temperature (solids fluctuating energy) can be calculated for each solid phase. You can select either an algebraic formulation, a constant, a user-defined function, or a partial differential equation.
  - Solid-phase shear and bulk viscosities are obtained by applying kinetic theory to granular flows. Frictional viscosity for modeling granular flow is also available. You can select appropriate models and user-defined functions for all properties.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

- Several interphase drag coefficient functions are available, which are appropriate for various types of multiphase regimes. (You can also modify the interphase drag coefficient through user-defined functions, as described in the Fluent Customization Manual.)
- All of the k - ε and k - ω turbulence models are available, and may apply to all phases or to the mixture.

14.5.2. Limitations of the Eulerian Model

All other features available in Ansys Fluent can be used in conjunction with the Eulerian multiphase model, except for the following limitations:

- The Reynolds Stress turbulence model is not available on a per phase basis.
- Particle tracking (using the Lagrangian dispersed phase model) interacts only with the primary phase.
- Streamwise periodic flow with specified mass flow rate cannot be modeled when the Eulerian model is used (you are allowed to specify a pressure drop).
- Inviscid flow is not allowed.
- Melting and solidification are not allowed.
- When tracking particles with the DPM model in combination with the Eulerian multiphase model, the **Shared Memory** method cannot be selected (Parallel Processing for the Discrete Phase Model). (Note that using the **Message Passing** or **Hybrid** method enables the compatibility of all multiphase flow models with the DPM model.)
- The Eulerian multiphase model is not compatible with non-premixed, partially premixed, and pre-mixed combustion models.

To change from a single-phase model to a multiphase model, you will have to do this in a series of steps. You will have to set up a mixture solution and then a multiphase solution. However, since multiphase problems are strongly linked, it is better to start directly solving a multiphase problem with an initial conservative set of parameters (first order in time and space). This is of course problem dependent. The modifications involve, among other things, the introduction of the volume fractions $\alpha_1, \alpha_2, ... \alpha_n$ for the multiple phases, as well as mechanisms for the exchange of momentum, heat, and mass between the phases.

14.5.3. Volume Fraction Equation

The description of multiphase flow as interpenetrating continua incorporates the concept of phasic volume fractions, denoted here by α_q. Volume fractions represent the space occupied by each phase, and the laws of conservation of mass and momentum are satisfied by each phase individually. The derivation of the conservation equations can be done by ensemble averaging the local instantaneous balance for each of the phases [18] (p. 1054) or by using the mixture theory approach.

The volume of phase q, V_q, is defined by

$$V_q = \int_V \alpha_q dV \tag{14.190}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

where

$$\sum_{q=1}^{n}\alpha_q=1 \tag{14.191}$$

The effective density of phase q is:

$$\hat{\rho}_q=\alpha_q\rho_q \tag{14.192}$$

where ρ_q is the physical density of phase q.

The volume fraction equation may be solved either through implicit or explicit time discretization. For detailed information about both VOF schemes, refer to The Implicit Formulation (p. 606) and The Explicit Formulation (p. 607).

14.5.4. Conservation Equations

The general conservation equations from which the equations solved by Ansys Fluent are derived are presented in this section, followed by the solved equations themselves.

14.5.4.1. Equations in General Form

14.5.4.1.1. Conservation of Mass

The continuity equation for phase q is

$$\frac{\partial}{\partial t}\left(\alpha_q\rho_q\right)+\nabla\cdot\left(\alpha_q\rho_q\vec{v}_q\right)=\sum_{p=1}^{n}\left(\dot{m}_{pq}-\dot{m}_{qp}\right)+S_q \tag{14.193}$$

where $\vec{v}_q$ is the velocity of phase q and $\dot{m}_{pq}$ characterizes the mass transfer from the p^{th} to q^{th} phase, and $\dot{m}_{qp}$ characterizes the mass transfer from phase q to phase p, and you are able to specify these mechanisms separately.

By default, the source term S_q on the right-hand side of Equation 14.193 (p. 662) is zero, but you can specify a constant or user-defined mass source for each phase. A similar term appears in the momentum and enthalpy equations. See Modeling Mass Transfer in Multiphase Flows (p. 751) for more information on the modeling of mass transfer in Ansys Fluent's general multiphase models.

14.5.4.1.2. Conservation of Momentum

The momentum balance for phase q yields

$$\begin{aligned}\frac{\partial}{\partial t}\left(\alpha_q\rho_q\vec{v}_q\right)+\nabla\cdot\left(\alpha_q\rho_q\vec{v}_q\vec{v}_q\right) &= -\alpha_q\nabla p+\nabla\cdot\bar{\bar{\tau}}_q+\alpha_q\rho_q\vec{g}\\ &+\sum_{p=1}^{n}\left(\vec{R}_{pq}+\dot{m}_{pq}\vec{v}_{pq}-\dot{m}_{qp}\vec{v}_{qp}\right)\\ &+\left(\vec{F}_q+\vec{F}_{lift,q}+\vec{F}_{wl,q}+\vec{F}_{vm,q}+\vec{F}_{td,q}\right)\end{aligned} \tag{14.194}$$

where $\bar{\bar{\tau}}_q$ is the q^{th} phase stress-strain tensor

$$\bar{\bar{\tau}}_q=\alpha_q\mu_q\left(\nabla\vec{v}_q+\nabla\vec{v}_q^{T}\right)+\alpha_q\left(\lambda_q-\frac{2}{3}\mu_q\right)\nabla\cdot\vec{v}_q\bar{\bar{I}} \tag{14.195}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Here μ_q and λ_q are the shear and bulk viscosity of phase q, $\vec{F}_q$ is an external body force, $\vec{F}_{lift,q}$ is a lift force (described in Lift Force (p. 682)), $\vec{F}_{wl,q}$ is a wall lubrication force (described in Wall Lubrication Force (p. 686)), $\vec{F}_{vm,q}$ is a virtual mass force, and $\vec{F}_{td,q}$ is a turbulent dispersion force (in the case of turbulent flows only). $\vec{R}_{pq}$ is an interaction force between phases, and p is the pressure shared by all phases.

$\vec{v}_{pq}$ is the interphase velocity, defined as follows. If $\dot{m}_{pq} > 0$ (that is, phase p mass is being transferred to phase q), $\vec{v}_{pq}=\vec{v}_p$; if $\dot{m}_{pq}<0$ (that is, phase q mass is being transferred to phase p), $\vec{v}_{pq}=\vec{v}_q$. Likewise, if $\dot{m}_{qp} > 0$ then $v_{qp}=v_q$, if $\dot{m}_{qp} < 0$ then $v_{qp}=v_p$.

Equation 14.194 (p. 662) must be closed with appropriate expressions for the interphase force $\vec{R}_{pq}$. This force depends on the friction, pressure, cohesion, and other effects, and is subject to the conditions that $\vec{R}_{pq}=-\vec{R}_{qp}$ and $\vec{R}_{qq}=0$.

Ansys Fluent uses a simple interaction term of the following form:

$$\sum_{p=1}^{n}\vec{R}_{pq}=\sum_{p=1}^{n}K_{pq}\left(\vec{v}_p-\vec{v}_q\right) \tag{14.196}$$

where K_{pq} ($=K_{qp}$) is the interphase momentum exchange coefficient (described in Interphase Exchange Coefficients (p. 667)), and $\vec{v}_p$ and $\vec{v}_q$ are the phase velocities. Note that Equation 14.196 (p. 663) represents the mean interphase momentum exchange and does not include any contribution due to turbulence. The turbulent interphase momentum exchange is modeled with the turbulent dispersion force term, $\vec{F}_{td,q}$ in Equation 14.194 (p. 662), as described in Turbulent Dispersion Force (p. 690).

14.5.4.1.3. Conservation of Energy

To describe the conservation of energy in Eulerian multiphase applications, a separate enthalpy equation for the q^{th} phase can be written as:

$$\begin{aligned}\frac{\partial}{\partial t}\left(\alpha_q\rho_q\left(e_q+\frac{\vec{v}_q^{\,2}}{2}\right)\right)+\nabla\cdot\left(\alpha_q\rho_q\vec{v}_q\left(h_q+\frac{\vec{v}_q^{\,2}}{2}\right)\right)&=\nabla\cdot\left(\alpha_q k_{eff,q}\nabla T_q-\sum_j h_{j,q}\vec{J}_{j,q}+\bar{\bar{\tau}}_{eff,q}\cdot\vec{v}_q\right)\\&\quad+\sum_{p=1}^{n}\left(Q_{pq}+\dot{m}_{pq}h_{pq}-\dot{m}_{qp}h_{qp}\right)+p\frac{\partial\alpha_q}{\partial t}+S_q\end{aligned} \tag{14.197}$$

where $k_{eff,q}$ is the effective conductivity, S_q is a source term that includes sources of energy (for example, due to chemical reaction or radiation), Q_{pq} is the intensity of heat exchange between the p^{th} and q^{th} phases. The heat exchange between phases must comply with the local balance conditions $Q_{pq}=-Q_{qp}$ and $Q_{qq}=0$. h_{pq} is the interphase enthalpy (for example, the enthalpy of the vapor at the temperature of the droplets, in the case of evaporation), $h_{j,q}$ is enthalpy of species j in phase q, and $\vec{J}_{j,q}$ is diffusive flux of species j in phase q.

In the above equation the energy and enthalpy of phase q are defined as follows; the enthalpy h_q is defined for ideal gas as:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$h_q = \sum_j Y_{j,q} h_{j,q} \tag{14.198}$$

and for an incompressible material includes a contribution from pressure work

$$h_q = \sum_j Y_{j,q} h_{j,q} + \frac{p}{\rho_q} \tag{14.199}$$

The sensible enthalpy of species $h_{j,q}$ is the part of enthalpy that includes only changes in the enthalpy due to specific heat

$$h_{j,q} = \int_{T_{ref}}^{T} c_{p,j,q} dT \tag{14.200}$$

The internal energy e_q is defined uniformly for compressible and incompressible materials as

$$e_q = h_q - \frac{p_{op} + p}{\rho_q} \tag{14.201}$$

In the above formulas p and p_{op} are the gauge and operating pressure, respectively. Such definitions of enthalpy and internal energy accomodate an incompressible ideal gas in common formulation

$$h = c_v T + \frac{p_{abs}}{\rho} = c_v T + R_g T + \frac{p}{\rho} = c_p T + \frac{p}{\rho} \tag{14.202}$$

14.5.4.2. Equations Solved by Ansys Fluent

The equations for fluid-fluid and granular multiphase flows, as solved by Ansys Fluent, are presented here for the general case of an n-phase flow.

14.5.4.2.1. Continuity Equation

The volume fraction of each phase is calculated from a continuity equation:

$$\frac{1}{\rho_{rq}} \left(\frac{\partial}{\partial t} \left(\alpha_q \rho_q \right) + \nabla \cdot \left(\alpha_q \rho_q \vec{v}_q \right) = \sum_{p=1}^{n} \left(\dot{m}_{pq} - \dot{m}_{qp} \right) \right) \tag{14.203}$$

where ρ_{rq} is the phase reference density, or the volume averaged density of the q^{th} phase in the solution domain.

The solution of this equation for each secondary phase, along with the condition that the volume fractions sum to one (given by Equation 14.191 (p. 662)), allows for the calculation of the primary-phase volume fraction. This treatment is common to fluid-fluid and granular flows.

14.5.4.2.2. Fluid-Fluid Momentum Equations

The conservation of momentum for a fluid phase q is

$$\begin{aligned} \frac{\partial}{\partial t} \left(\alpha_q \rho_q \vec{v}_q \right) + \nabla \cdot \left(\alpha_q \rho_q \vec{v}_q \vec{v}_q \right) = & -\alpha_q \nabla p + \nabla \cdot \bar{\bar{\tau}}_q + \alpha_q \rho_q \vec{g} \\ & + \sum_{p=1}^{n} \left(K_{pq} \left(\vec{v}_p - \vec{v}_q \right) + \dot{m}_{pq} \vec{v}_{pq} - \dot{m}_{qp} \vec{v}_{qp} \right) \\ & + \left(\vec{F}_q + \vec{F}_{lift,q} + \vec{F}_{wl,q} + \vec{F}_{vm,q} + \vec{F}_{td,q} \right) \end{aligned} \tag{14.204}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Here $\vec{g}$ is the acceleration due to gravity and $\overline{\overline{\tau}}_q$, $\vec{F}_q$, $\vec{F}_{lift,q}$, $\vec{F}_{wl,q}$, $\vec{F}_{vm,q}$, and $\vec{F}_{td,q}$ are as defined for Equation 14.194 (p. 662).

14.5.4.2.3. Fluid-Solid Momentum Equations

Following the work of [14] (p. 1053) , [99] (p. 1058) , [142] (p. 1061) , [205] (p. 1064) , [345] (p. 1073) , [394] (p. 1075) , [482] (p. 1081) , [637] (p. 1090) , Ansys Fluent uses a multi-fluid granular model to describe the flow behavior of a fluid-solid mixture. The solid-phase stresses are derived by making an analogy between the random particle motion arising from particle-particle collisions and the thermal motion of molecules in a gas, taking into account the inelasticity of the granular phase. As is the case for a gas, the intensity of the particle velocity fluctuations determines the stresses, viscosity, and pressure of the solid phase. The kinetic energy associated with the particle velocity fluctuations is represented by a "pseudothermal" or granular temperature which is proportional to the mean square of the random motion of particles.

The conservation of momentum for the fluid phases is similar to Equation 14.204 (p. 664) , and that for the s^{th} solid phase is

$$\begin{aligned}\frac{\partial}{\partial t}(\alpha_s\rho_s\vec{v}_s)+\nabla\cdot(\alpha_s\rho_s\vec{v}_s\vec{v}_s) = &-\alpha_s\nabla p-\nabla p_s+\nabla\cdot\overline{\overline{\tau}}_s+\alpha_s\rho_s\vec{g}\\ &+\sum_{l=1}^{N}\left(K_{ls}(\vec{v}_l-\vec{v}_s)+\dot{m}_{ls}\vec{v}_{ls}-\dot{m}_{sl}\vec{v}_{sl}\right)\\ &+\left(\vec{F}_s+\vec{F}_{lift,s}+\vec{F}_{vm,s}+\vec{F}_{td,s}\right)\end{aligned} \tag{14.205}$$

where p_s is the s^{th} solids pressure, $K_{ls}=K_{sl}$ is the momentum exchange coefficient between fluid or solid phase l and solid phase s, N is the total number of phases, and $\vec{F}_s$, $\vec{F}_{lift,s}$, $\vec{F}_{vm,s}$, and $\vec{F}_{td,s}$ are defined in the same manner as the analogous terms in Equation 14.194 (p. 662).

14.5.4.2.4. Conservation of Energy

The equation solved by Ansys Fluent for the conservation of energy is Equation 14.197 (p. 663).

14.5.5. Surface Tension and Adhesion for the Eulerian Multiphase Model

The effects of surface tension and wall adhesion along the interface between a pair of phases can be included in your Eulerian multiphase simulation. See Surface Tension and Adhesion (p. 613) for details on how they are calculated.

14.5.6. Interfacial Area Concentration

Interfacial area concentration is defined as the interfacial area between two phases per unit mixture volume. This is an important parameter for predicting mass, momentum and energy transfers through the interface between the phases. When using the Eulerian multiphase model with non-granular secondary phases, you can have Ansys Fluent compute the interfacial area in one of two ways:

- use a transport equation for interfacial area concentration as described in Interfacial Area Concentration (p. 647). This allows for a distribution of bubble diameters and coalescence/breakage effects.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

- use an algebraic relationship between a specified bubble diameter and the interfacial area density.

The algebraic interfacial area concentration models are derived from the surface area to volume ratio, A_p , for a spherical bubble or droplet:

$$A_p = \frac{\pi d_p^2}{\frac{1}{6}\pi d_p^3} = \frac{6}{d_p} \tag{14.206}$$

where d_p is the bubble or droplet diameter. The algebraic models available when using the Eulerian multiphase model are:

- Particle model (default)

  For a dispersed phase, p, with volume fraction, α_p, the particle model estimates the interfacial area density, A_i as

  $$A_i = \alpha_p A_p = \frac{6\alpha_p}{d_p} \tag{14.207}$$

- Symmetric model

  The symmetric model treats both phases p and q symmetrically. Phases p and q can be continuous or dispersed. The interphase area density is calculated as:

  $$A_i = \frac{6\alpha_q \alpha_p}{d_{p,q}} \tag{14.208}$$

  where α_p and α_q are the volume fraction of the phase p and phase q, respectively, and $d_{p,q}$ is the characteristic length scale computed as follows:

  – If the phase p is a dispersed phase:

  $$d_{p,q} = d_p \tag{14.209}$$

  – If both phases p and q are dispersed:

  $$d_{p,q} = \alpha_q d_p + \alpha_p d_q \tag{14.210}$$

- Ishii model (boiling flows only)

  The Ishii model, which is only available when the boiling model is activated, also modifies the particle model and results in a piecewise linear function of α_p that approaches 0 as α_p approaches 1.

  $$A_i = \frac{6(1-\alpha_p)\min(\alpha_p, \alpha_{p,crit})}{d_p(1-\min(\alpha_p, \alpha_{p,crit}))} \tag{14.211}$$

  In Fluent, $\alpha_{p,crit}$ is chosen as 0.25.

- Gradient model

  For the description of the Gradient model, refer to Algebraic Models (p. 651).

- User-Defined (boiling flows only)

  See `DEFINE_EXCHANGE_PROPERTY` in the Fluent Customization Manual.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

14.5.7. Interphase Exchange Coefficients

It can be seen in Equation 14.204 (p. 664) and Equation 14.205 (p. 665) that momentum exchange between the phases is based on the value of the fluid-fluid exchange coefficient K_{pq} and, for granular flows, the fluid-solid and solid-solid exchange coefficients K_{ls}.

Note:

Note that in Ansys Fluent, all the available interphase exchange coefficient models are empirically based. At the present, there is no general formulation in the literature, and attention is required in some situations, such as:

- air / liquid / solids configurations
- polydispersed flow
- porous media
- compressible flows
- temperature variation
- dense situations
- near-wall flows

The interphase exchange coefficients for such cases may need modification, which can be dealt with using user-defined functions.

14.5.7.1. Fluid-Fluid Exchange Coefficient

For fluid-fluid flows, each secondary phase is assumed to form droplets or bubbles. This has an impact on how each of the fluids is assigned to a particular phase. For example, in flows where there are unequal amounts of two fluids, the predominant fluid should be modeled as the primary fluid, since the sparser fluid is more likely to form droplets or bubbles. The exchange coefficient for these types of bubbly, liquid-liquid or gas-liquid mixtures can be written in the following general form:

$$K_{pq}=\frac{\rho_p f}{6\tau_p} d_p A_i \tag{14.212}$$

where A_i is the interfacial area (see Interfacial Area Concentration (p. 665)), f, the drag function, is defined differently for the different exchange-coefficient models (as described below) and τ_p, the "particulate relaxation time", is defined as

$$\tau_p=\frac{\rho_p d_p^2}{18\mu_q} \tag{14.213}$$

where d_p is the diameter of the bubbles or droplets of phase p.

Nearly all definitions of f include a drag coefficient (C_D) that is based on the relative Reynolds number (Re): It is this drag function that differs among the exchange-coefficient models. For all

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

these situations, K_{pq} should tend to zero whenever the primary phase is not present within the domain.

For each pair of phases in fluid-fluid flows you may use one of the available drag function models in Ansys Fluent or a user-defined function to specify the interphase exchange coefficient. If the exchange coefficient is equal to zero (that is, if no exchange coefficient is specified), the flow fields for the fluids will be computed independently, with the only "interaction" being their complementary volume fractions within each computational cell. You can specify different exchange coefficients for each pair of phases.

The following drag models are available in Ansys Fluent.

14.5.7.1.1. Schiller and Naumann Model

14.5.7.1.2. Morsi and Alexander Model

14.5.7.1.3. Symmetric Model

14.5.7.1.4. Grace et al. Model

14.5.7.1.5. Tomiyama et al. Model

14.5.7.1.6. Ishii Model

14.5.7.1.7. Ishii-Zuber Drag Model

14.5.7.1.8. Universal Drag Laws for Bubble-Liquid and Droplet-Gas Flows

14.5.7.1.1. Schiller and Naumann Model

For the model of Schiller and Naumann [574] (p. 1086),

$$f=\frac{C_D Re}{24} \tag{14.214}$$

where

$$C_D=\begin{cases}24\left(1+0.15Re^{0.687}\right)/Re & Re\leq 1000\\ 0.44 & Re>1000\end{cases} \tag{14.215}$$

and Re is the relative Reynolds number. The relative Reynolds number for the primary phase q and secondary phase p is obtained from

$$Re=\frac{\rho_q\left|\vec{v}_p-\vec{v}_q\right|d_p}{\mu_q} \tag{14.216}$$

The relative Reynolds number for secondary phases p and r is obtained from

$$Re=\frac{\rho_{rp}\left|\vec{v}_r-\vec{v}_p\right|d_{rp}}{\mu_{rp}} \tag{14.217}$$

where $\mu_{rp}=\alpha_p\mu_p+\alpha_r\mu_r$ is the mixture viscosity of the phases p and r.

The Schiller and Naumann model is the default method, and it is acceptable for general use for all fluid-fluid pairs of phases.

14.5.7.1.2. Morsi and Alexander Model

For the Morsi and Alexander model [458] (p. 1079) :

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$f=\frac{C_D Re}{24} \tag{14.218}$$

where

$$C_D=a_1+\frac{a_2}{Re}+\frac{a_3}{Re^2} \tag{14.219}$$

and Re is defined by Equation 14.216 (p. 668) or Equation 14.217 (p. 668). The constants, a_i, are defined as follows:

$$a_1,a_2,a_3=\begin{cases} 0,24,0 & 0<Re<0.1 \\ 3.690,22.73,0.0903 & 0.1<Re<1 \\ 1.222,29.1667,-3.8889 & 1<Re<10 \\ 0.6167,46.50,-116.67 & 10<Re<100 \\ 0.3644,98.33,-2778 & 100<Re<1000 \\ 0.357,148.62,-47500 & 1000<Re<5000 \\ 0.46,-490.546,578700 & 5000<Re<10000 \\ 0.5191,-1662.5,5416700 & Re\geq 10000 \end{cases} \tag{14.220}$$

The Morsi and Alexander model is the most complete, adjusting the function definition frequently over a large range of Reynolds numbers, but calculations with this model may be less stable than with the other models.

14.5.7.1.3. Symmetric Model

For the symmetric model, the density and the viscosity are calculated from volume averaged properties:

$$\rho_{pq}=\alpha_p\rho_p+\alpha_q\rho_q \tag{14.221}$$

$$\mu_{pq}=\alpha_p\mu_p+\alpha_q\mu_q \tag{14.222}$$

and the diameter is defined as

$$d_{pq}=\frac{1}{2}\left(d_p+d_q\right) \tag{14.223}$$

in turn

$$K_{pq}=\frac{\rho_{pq}f}{6\tau_{pq}}d_pA_i \tag{14.224}$$

$$\tau_{pq}=\frac{\rho_{pq}\left(d_{pq}\right)^2}{18\mu_{pq}} \tag{14.225}$$

$$Re=\frac{\varrho_{pq}\left|v_p-v_q\right|d_{pq}}{\mu_{pq}} \tag{14.226}$$

$$f=\frac{C_D Re}{24} \tag{14.227}$$

and C_D is defined by Equation 14.215 (p. 668).

Note that if there is only one dispersed phase, then $d_p=d_q$ in Equation 14.223 (p. 669).

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

The symmetric model is recommended for flows in which the secondary (dispersed) phase in one region of the domain becomes the primary (continuous) phase in another. For example, if air is injected into the bottom of a container filled halfway with water, the air is the dispersed phase in the bottom half of the container; in the top half of the container, the air is the continuous phase. This model can also be used for the interaction between secondary phases.

14.5.7.1.4. Grace et al. Model

The Grace et al. model is well suited to gas-liquid flows in which the bubbles can have a range of shapes.

For the model of Grace, et al. [111] (p. 1059) :

$$f = \frac{C_D Re}{24} \tag{14.228}$$

where

$$Re = \frac{\rho_q \left|\vec{v}_p - \vec{v}_q\right| d_p}{\mu_q} \tag{14.229}$$

$$C_D = \alpha_q^{C_{exp}} \cdot \max\left(\min\left(C_{D_{ellipse}}, C_{D_{cap}}\right), C_{D_{sphere}}\right) \tag{14.230}$$

In Equation 14.230 (p. 670), α_q is the volume fraction of the continuous phase; C_{exp} is the volume fraction correction exponent; and $C_{D_{ellipse}}$, $C_{D_{cap}}$, and $C_{D_{sphere}}$ are defined as:

$$C_{D_{sphere}} = \begin{cases} 24/Re & Re < 0.01 \\ 24\left(1 + 0.15 Re^{0.687}\right)/Re & Re \geq 0.01 \end{cases} \tag{14.231}$$

$$C_{D_{cap}} = \frac{8}{3} \tag{14.232}$$

$$C_{D_{ellipse}} = \frac{4}{3} \frac{g d_p}{U_t^2} \frac{\left(\rho_q - \rho_p\right)}{\rho_q} \tag{14.233}$$

where

$$U_t = \frac{\mu_q}{\rho_q d_p} Mo^{-0.149} \left(J - 0.857\right)$$

where Mo is the Morton number given by:

$$Mo = \frac{\mu_q^4 g \left(\rho_q - \rho_p\right)}{\rho_q^2 \sigma^3}$$

J is given by the piecewise function:

$$J = \begin{cases} 0.94 H^{0.757} & 2 < H \leq 59.3 \\ 3.42 H^{0.441} & H > 59.3 \end{cases}$$

$$H = \frac{4}{3} Eo Mo^{-0.149} \left(\frac{\mu_q}{\mu_{ref}}\right)^{-0.14}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

where Eo is the Eötvös number:

$$Eo=\frac{g(\rho_q-\rho_p)d_p^2}{\sigma}$$

and $\mu_{ref}=0.0009 kg/(m-s)$.

- Sparsely distributed fluid particles

  In flows with sparsely distributed fluid particles, C_{exp} in Equation 14.230 (p. 670) is zero.

- Densely distributed fluid particles

  For high bubble volume fractions, a non-zero value for C_{exp} should be used depending on the bubble size.

See Specifying the Drag Function in the *Fluent User's Guide* for more information.

14.5.7.1.5. Tomiyama et al. Model

For the model of Tomiyama, et al. [640] (p. 1090)

$$f=\frac{C_D Re}{24} \tag{14.234}$$

where

$$Re=\frac{\rho_q|\vec{v}_p-\vec{v}_q|d_p}{\mu_q} \tag{14.235}$$

$$C_D=\max\left(\min\left(\frac{24}{Re}\left(1+0.15Re^{0.687}\right),\frac{72}{Re}\right),\frac{8}{3}\frac{Eo}{Eo+4}\right) \tag{14.236}$$

In Equation 14.236 (p. 671),

$$Eo=\frac{g(\rho_q-\rho_p)d_p^2}{\sigma}$$

Like the Grace et al. model, the Tomiyama et al. model is well suited to gas-liquid flows in which the bubbles can have a range of shapes.

14.5.7.1.6. Ishii Model

For boiling flows only, you can use the model of Ishii [269] (p. 1068). For the Ishii model, the drag coefficient C_D is determined by choosing the minimum of the viscous regime $C_{D_{vis}}$ and the distorted regime $C_{D_{dis}}$, defined as follows:

$$C_D=\min\left(C_{D_{vis}},C_{D_{dis}}\right) \tag{14.237}$$

where $C_{D_{vis}}$ and $C_{D_{dis}}$ are given by the following formulas:

$$C_{D_{vis}}=\frac{24}{\mathrm{Re}}\left(1+0.15\mathrm{Re}^{0.75}\right) \tag{14.238}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$C_{D_{dis}} = \frac{2}{3}\frac{d_p}{\sqrt{\frac{\sigma}{g|\rho_q - \rho_p|}}} \tag{14.239}$$

where Re is the relative Reynolds number, σ is the surface tension, and g is gravity.

The bubble diameter d_p is determined as in Bubble and Droplet Diameters (p. 740).

14.5.7.1.7. Ishii-Zuber Drag Model

The Ansys Fluent implementation of the Ishii-Zuber drag coefficient automatically accounts for different particle distribution regimes.

- Sparsely Distributed Fluid Particles

  For very small Reynolds numbers, fluid particles (bubbles or droplets) tend to behave as solid spherical particles. In this case, the drag coefficient can be accurately approximated by the Schiller-Naumann correlation (Equation 14.214 (p. 668).

  As the particle Reynolds number increases, the effect of surface tension becomes more important in what is known as the inertial or distorted particle regime. The particles at first become distorted into ellipsoids, and then take the shape of a spherical cap.

  – Distorted Regime

    In the distorted particle regime, the drag coefficient is independent of Reynolds number, but highly dependent on particle shape. To calculate the drag coefficient, the Ishii-Zuber model uses the Eotvos number, which measures the ratio between gravitational and surface tension forces:

    $$Eo = \frac{g\Delta\rho d_p^2}{\sigma} \tag{14.240}$$

    where $\Delta\rho$ is the density difference between the phases, g is the gravitational acceleration, and σ is the surface tension coefficient between each phase pair.

    For the ellipsoidal fluid particles, the drag coefficient is calculated by:

    $$C_{ellipse} = \frac{2}{3}\sqrt{Eo} \tag{14.241}$$

  – Spherical Cap Regime

    In the spherical cap regime, the drag coefficient is approximated by:

    $$C_{D_{cap}} = \frac{8}{3} \tag{14.242}$$

- Densely Distributed Fluid Particles

  The correlations Equation 14.241 (p. 672) and Equation 14.242 (p. 672) are valid for flows in which fluid particles move in a sparsely distributed fashion. However, these correlations are not applicable for high void fraction flows. As the volume fraction of the gas increases, bubbles begin to accumulate and move as a swarm of particles, thus modifying the effective drag. The Ishii-Zuber drag model automatically accounts for dense particle effects by using a swarm factor correction.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

– Viscous Regime

In the viscous regime, where fluid particles can be considered spherical, the Schiller Naumann correlation Equation 14.214 (p. 668) is modified using the mixture Reynolds number based on a mixture viscosity as follows:

$$\begin{aligned} C_{D_{sphere}} &= \frac{24}{Re_m}\left(1+0.15\ Re_m^{0.687}\right) \\ Re_m &= \frac{\rho_c \left|\vec{V_p}-\vec{V_q}\right| d_p}{\mu_m} \\ \frac{\mu_m}{\mu_q} &= \left(1-\alpha_p\right)^{-2.5\,\mu_*} \\ \mu_* &= \frac{\mu_p+0.4\,\mu_q}{\mu_p+\mu_q} \end{aligned} \tag{14.243}$$

– Distorted Regime

In the distorted particle regime, the single particle drag coefficient is multiplied by the Ishii-Zuber swarm factor correction:

$$\begin{aligned} C_{D_{ellipse}} &= E_\alpha\, C_{D_\infty} \\ C_{D_\infty} &= \frac{2}{3}\ Eo^{1/2} \\ E_\alpha &= \left[\frac{\left(1+17.67\, f_\alpha^{6/7}\right)}{18.67\, f_\alpha}\right]^2 \\ f_\alpha &= \frac{\mu_q}{\mu_m}\left(1-\alpha_p\right)^{1/2} \end{aligned} \tag{14.244}$$

– Spherical Cap Regime

$$\begin{aligned} C_{D_{cap}} &= C_{D_\infty}\ \left(1-\alpha_p\right)^2 \\ C_{D_\infty} &= \frac{8}{3} \end{aligned} \tag{14.245}$$

Ansys Fluent automatically calculates the drag coefficient as:

$$C_D = \begin{cases} C_{D_{sphere}}, & if\ C_{D_{sphere}} \geq C_{D_{ellipse}} \\ \min\left(C_{D_{ellipse}}, C_{D_{cap}}\right) & if\ C_{D_{sphere}} < C_{D_{ellipse}} \end{cases} \tag{14.246}$$

14.5.7.1.8. Universal Drag Laws for Bubble-Liquid and Droplet-Gas Flows

The universal drag laws [317] (p. 1071) are suitable for the calculation of the drag coefficients in a variety of gas-liquid flow regimes. The drag laws can apply to non-spherical droplets/bubbles with the constraint of a pool flow regime, that is, the hydraulic diameter of the flow domain which is far larger than the averaged size of the particles.

The exchange coefficient for bubbly and droplet flows can be written in the general form

$$K_{pq} = \frac{\rho_p f}{6\tau_p} d_p A_i \tag{14.247}$$

Where q represents the primary phase and p the dispersed phase and A_i is the interfacial area (see Interfacial Area Concentration (p. 665)). The dispersed phase relaxation time τ_p is defined as

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$\tau_p = \frac{\rho_p d_p^2}{18\mu_e} \tag{14.248}$$

The drag function f is defined as

$$f = \frac{C_D Re}{24} \tag{14.249}$$

The relative Reynolds number for the primary phase q and the secondary phase p is obtained based on the relative velocity of the two phases.

$$Re = \frac{\rho_q |\vec{v}_q - \vec{v}_p| d_p}{\mu_e} \tag{14.250}$$

Where μ_e is the effective viscosity of the primary phase accounting for the effects of family of particles in the continuum.

The Rayleigh-Taylor instability wavelength is

$$\lambda_{RT} = \left(\frac{\sigma}{g \Delta \rho_{pq}} \right)^{0.5} \tag{14.251}$$

Where σ is the surface tension, g the gravity, and $\Delta \rho_{pq}$ the absolute value of the density difference between phases p and q.

The drag coefficient is defined differently for bubbly and droplet flows.

14.5.7.1.8.1. Bubble-Liquid Flow

$$C_{D_{vis}} = \frac{24}{Re}\left(1 + 0.1 Re^{0.75}\right) \tag{14.252}$$

$$C_{D_{dis}} = 2/3 \left(\frac{d_p}{\lambda_{RT}} \right) \left\{ \frac{1 + 17.67 f^{*\,6/7}}{18.67 f^*} \right\}^2 ; \quad f^* = \left(1 - \alpha_p\right)^{1.5} \tag{14.253}$$

$$C_{D_{cap}} = \frac{8}{3}\left(1 - \alpha_p\right)^2 \tag{14.254}$$

- In the viscous regime, the following condition is satisfied:

  $$C_{D_{dis}} < C_{D_{vis}} \tag{14.255}$$

  The drag coefficient, C_D, is defined as

  $$C_D = C_{D_{vis}} \tag{14.256}$$

- In the distorted bubble regime, the following condition is satisfied:

  $$C_{D_{vis}} < C_{D_{dis}} < C_{D_{cap}} \tag{14.257}$$

  The drag coefficient is calculated as

  $$C_D = C_{D_{dis}} \tag{14.258}$$

- In the strongly deformed, capped bubble regime, the following condition is satisfied:

  $$C_{D_{dis}} > C_{D_{cap}} \tag{14.259}$$

  The drag coefficient can be written as

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$C_D = C_{D_{cap}} \tag{14.260}$$

The effective viscosity for the bubble-liquid mixture is

$$\mu_e = \frac{\mu_q}{1-\alpha_p} \tag{14.261}$$

14.5.7.1.8.2. Droplet-Gas Flow

- When $\mathrm{Re} < 1$, the drag coefficient for the stokes regime is

$$C_D = \frac{24}{Re} \tag{14.262}$$

- When $1 \le \mathrm{Re} \le 1000$, the drag coefficient for the viscous regime is

$$C_D = \frac{24}{Re}\left(1+0.1Re^{0.75}\right) \tag{14.263}$$

- For the Newton's regime (Re ≥ 1000), the drag coefficient is

$$C_D = 2/3\left(\frac{d_p}{\lambda_{RT}}\right)\left\{\frac{1+17.67f^{*\,6/7}}{18.67f^*}\right\}^2 ;\quad f^* = \left(1-\alpha_p\right)^3 \tag{14.264}$$

The effective viscosity for a droplet-gas mixture is

$$\mu_e = \frac{\mu_q}{\left(1-\alpha_p\right)^{2.5}} \tag{14.265}$$

Important:

The universal drag models are currently suitable for bubble-liquid and/or droplet-gas flow where the characteristic length of the flow domain is much larger than the averaged size of the particles.

14.5.7.2. Fluid-Solid Exchange Coefficient

The fluid-solid exchange coefficient K_{sl} can be written in the following general form:

$$K_{sl} = \frac{\alpha_s \rho_s f}{\tau_s} \tag{14.266}$$

where α_s is the volume fraction of the solid phase, and ρ_s is the density of the solid phase. f is defined differently for the different exchange-coefficient models (as described below), and τ_s , the "particulate relaxation time", is defined as

$$\tau_s = \frac{\rho_s d_s^2}{18\mu_l} \tag{14.267}$$

where d_s is the diameter of particles of phase s, and μ_l is the dynamic viscosity of fluid phase.

Here and below, the subscripts l and s denote the l^{th} fluid phase and the s^{th} solid phase, respectively.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

All definitions of f include a drag function (C_D) that is based on the relative Reynolds number (Re_s). It is this drag function that differs among the exchange-coefficient models.

- Syamlal-O'Brien model

  For the Syamlal-O'Brien model [636] (p. 1090):

$$f=\frac{C_D Re_s \alpha_l}{24 v_{r,s}^2} \tag{14.268}$$

  where α_l is the volume fraction of fluid phase, and the drag function has a form derived by Dalla Valle [131] (p. 1060):

$$C_D=\left(0.63+\frac{4.8}{\sqrt{Re_s/v_{r,s}}}\right)^2 \tag{14.269}$$

  This model is based on measurements of the terminal velocities of particles in fluidized or settling beds, with correlations that are a function of the volume fraction and relative Reynolds number [554] (p. 1085):

$$Re_s=\frac{\rho_l d_s \left|\vec{v}_s-\vec{v}_l\right|}{\mu_l} \tag{14.270}$$

  where $\vec{v}_s$ and $\vec{v}_l$ are the velocities of solid and liquid phases, respectively, ρ_l is the density of the fluid phase, and d_s is the diameter of the s^{th} solid phase particles.

  The fluid-solid exchange coefficient has the form

$$K_{sl}=\frac{3\alpha_s \alpha_l \rho_l}{4 v_{r,s}^2 d_s} C_D\left(\frac{Re_s}{v_{r,s}}\right)\left|\vec{v}_s-\vec{v}_l\right| \tag{14.271}$$

  where $v_{r,s}$ is the terminal velocity correlation for the solid phase [197] (p. 1064):

$$v_{r,s}=0.5\left(A-0.06Re_s+\sqrt{(0.06Re_s)^2+0.12Re_s(2B-A)+A^2}\right) \tag{14.272}$$

  with

$$A=\alpha_l^{4.14} \tag{14.273}$$

  and

$$B=0.8\alpha_l^{1.28} \tag{14.274}$$

  for $\alpha_l \leq 0.85$, and

$$B=\alpha_l^{2.65} \tag{14.275}$$

  for $\alpha_l > 0.85$.

  This model is appropriate when the solids shear stresses are defined according to Syamlal et al. [637] (p. 1090) (Equation 14.364 (p. 697)).

- Parametrized Syamlal-O'Brien model

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

The parametrized Syamlal-O'Brien model is an enhancement of the Syamlal-O'Brien model in which the values of 0.8 and 2.65 in Equation 14.274 (p. 676) and Equation 14.275 (p. 676) are replaced by parameters that are adjusted based on the fluid flow properties and the expected minimum fluidization velocity [635] (p. 1089). This overcomes the tendency of the original Syamlal-O'Brien model to under/over-predict bed expansion in fluid bed reactors, for example.

The parameters are derived from the velocity correlation between single and multiple particle systems at terminal settling or minimum fluidization conditions. For a multiple particle system, the relative Reynolds number at the minimum fluidization condition is expressed as:

$$\mathrm{Re}_s = \mathrm{Re}_{single}\left[\frac{A + 0.06 B \mathrm{Re}_{single}}{1 + 0.06 \mathrm{Re}_{single}}\right] \tag{14.276}$$

where Re_{single} is the Reynolds number at the terminal settling condition for a single particle and can be expressed as:

$$\mathrm{Re}_{single} = \left[\frac{\left(4.8^2 + 2.52\sqrt{4Ar/s}\right)^{0.5} - 4.8}{1.26}\right]^2 \tag{14.277}$$

The Archimedes number, Ar, can be written as a function of the drag coefficient, C_D, and the Reynolds number, Re_s:

$$\mathrm{Ar} = \frac{3}{4} C_D \mathrm{Re}_s^2 \tag{14.278}$$

where C_D can be obtained from the terminal velocity correlation, $v_{r,s}$, and the Reynolds number:

$$C_D = \left(\frac{0.63}{v_{r,s}} + \frac{4.8}{\sqrt{v_{r,s}\mathrm{Re}_s}}\right)^2 \tag{14.279}$$

$v_{r,s}$ is found from Equation 14.272 (p. 676) with A from Equation 14.273 (p. 676) and B rewritten as follows:

$$B = \begin{cases} c_1 \alpha_l^{1.28} & \alpha_l \le 0.85 \\ \alpha_l^{d_1} & \alpha_l > 0.85 \end{cases} \tag{14.280}$$

where $d_1 = 1.28 + \dfrac{\log_{10} c_1}{\log_{10} 0.85}$

Once the particle diameter and the expected minimum fluidization velocity are given, the coefficients c_1 and d_1 can be found by iteratively solving Equation 14.276 (p. 677) – Equation 14.280 (p. 677).

This model implementation is restricted to use in gas-solid flows in which the gas phase is the primary phase and is incompressible. Furthermore, the model is appropriate only for Geldart Group B particles.

- Wen and Yu [697] (p. 1093) model

  For the model of Wen and Yu [697] (p. 1093), the fluid-solid exchange coefficient is of the following form:

$$K_{sl} = \frac{3}{4} C_D \frac{\alpha_s \alpha_l \rho_l \left|\vec{v}_s - \vec{v}_l\right|}{d_s} \alpha_l^{-2.65} \tag{14.281}$$

  where

$$C_D=\frac{24}{\alpha_l Re_s}\left[1+0.15\left(\alpha_l Re_s\right)^{0.687}\right] \tag{14.282}$$

and Re_s is defined by Equation 14.270 (p. 676).

This model is appropriate for dilute systems.

- Gidaspow model

  The Gidaspow model [205] (p. 1064) is a combination of the Wen and Yu model [697] (p. 1093) and the Ergun equation [164] (p. 1062).

  When $\alpha_l > 0.8$, the fluid-solid exchange coefficient K_{sl} is of the following form:

$$K_{sl}=\frac{3}{4}C_D\frac{\alpha_s\alpha_l\rho_l\left|\vec{v}_s-\vec{v}_l\right|}{d_s}\alpha_l^{-2.65} \tag{14.283}$$

  where

$$C_D=\frac{24}{\alpha_l Re_s}\left[1+0.15\left(\alpha_l Re_s\right)^{0.687}\right] \tag{14.284}$$

  When $\alpha_l \leq 0.8$,

$$K_{sl}=150\frac{\alpha_s\left(1-\alpha_l\right)\mu_l}{\alpha_l d_s^2}+1.75\frac{\rho_l\alpha_s\left|\vec{v}_s-\vec{v}_l\right|}{d_s} \tag{14.285}$$

  This model is recommended for dense fluidized beds.

- Huilin-Gidaspow model

  The Huilin-Gidaspow model [262] (p. 1068) is also a combination of the Wen and Yu model [697] (p. 1093) and the Ergun equation [164] (p. 1062). The smooth switch is provided by the function when the solid volume fraction is less than 0.2:

$$K_{sl}=\psi K_{sl-Ergun}+\left(1-\psi\right)K_{sl-Wen\&Yu} \tag{14.286}$$

  where the stitching function is of the form

$$\psi=\frac{1}{2}+\frac{\arctan\left(262.5\left(\alpha_s-0.2\right)\right)}{\pi} \tag{14.287}$$

- Gibilaro model

  The Gibilaro model [202] (p. 1064) is of the form

$$K_{sl}=\left(\frac{18}{\mathrm{Re}}+0.33\right)\frac{\rho_f\left|v_s-v_l\right|}{d_p}\alpha_s\alpha_l^{-1.8} \tag{14.288}$$

  with the Reynolds number as

$$\mathrm{Re}=\frac{\alpha_l\rho_l d_p\left|v_s-v_l\right|}{\mu_l} \tag{14.289}$$

- EMMS model

  The energy-minimization multi-scale (EMMS) drag model is a heterogeneous approach derived from the mesoscale-structure-based methods. In the EMMS method, the mesoscale structures

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

are broken down into a cluster phase and a dilute phase ([687] (p. 1093), [388] (p. 1075)). While homogeneous drag laws (such as Wen and Yu model [697] (p. 1093) and Gidaspow model [205] (p. 1064)) have the tendency to over-predict the solids flux, the cluster-based EMMS drag correctly evaluates the solid flux and axially S-shaped profiles of voidage. The EMMS drag model proposed by Lu et al. [388] (p. 1075) is suitable for modeling two-phase granular flow in fluidized beds.

The fluid-solid exchange coefficient is of the following form:

$$K_{sl}=\frac{3}{4}C_D\frac{\alpha_s\alpha_l\rho_l|\vec{v}_s-\vec{v}_l|}{d_s}\alpha_l^{-265}H_D \tag{14.290}$$

where

$$C_D=\begin{cases}0.44, & Re_S\geq 1000\\ \frac{24}{Re_S}\left(1+0.15Re_S^{0.687}\right), & Re_S<1000\end{cases}$$

where Re_S is defined by

$$Re_S=\frac{\rho_l d_s|\vec{v}_s-\vec{v}_l|}{\mu_l}\alpha_l$$

Index H_D is expressed as:

$$H_D=a\left(Re_S+b\right)^c,\quad 0.001\leq Re_S\leq 35 \tag{14.291}$$

where the coefficients a, b, and c are functions of gas phase volume fraction. Formulas for calculating a, b, and c over different ranges of α_l are shown in Table 14.3: Fitting Formulas for Index H_D (p. 679) [388] (p. 1075).

Table 14.3: Fitting Formulas for Index H_D

Range of α_l	Coefficients
$0.4\leq \alpha_l <0.46$	$\begin{cases}a=0.8526-\frac{0.5846}{1+(\alpha_l/0.4325)^{22.6279}}\\ c=0\end{cases}$
$0.46\leq \alpha_l <0.545$	$\begin{cases}a=0.032+\frac{0.7399}{1+(\alpha_l/0.4912)^{54.4265}}\\ b=0.00225+\frac{772.0074}{1+10^{66.3224(\alpha_l-0.3987)}}+\frac{0.02404}{1+10^{53.8948(0.5257-\alpha_l)}}\\ c=0.1705-\frac{0.1731}{1+(\alpha_l/0.502)^{37.7091}}\end{cases}$
$0.545\leq \alpha_l <0.99$	$\begin{cases}a=(2124.956-2142.349\,\alpha_l)^{-0.4896}\\ b=(0.8223-0.1293\alpha_l)^{13.031}\\ c=\frac{(\alpha_l-1.0013)}{-0.06633+9.1391(\alpha_l-1.0013)+6.9231(\alpha_l-1.0013)^2}\end{cases}$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Range of α_l	Coefficients
$0.99 \leq \alpha_l < 0.9997$	$\begin{cases} a = 0.4243 + \frac{0.88}{1+\exp\left(-(\alpha_l - 0.9942)/0.00218\right)}\left(1 - \frac{1}{1+\exp\left(-\frac{\alpha_l - 0.9989}{0.00003}\right)}\right) \\ b = 0.01661 + 0.2436\exp\left(-0.5\left(\frac{\alpha_l - 0.9985}{0.00191}\right)^2\right) \\ c = 0.0825 - 0.0574\exp\left(-0.5\left(\frac{\alpha_l - 0.9979}{0.00703}\right)^2\right) \end{cases}$
$0.9997 \leq \alpha_l < 1$	$a = 1$ $c = 1$

- Filtered two-fluid model

  The filtered two-fluid model can be used for cases with coarse meshes as described in The Filtered Two-Fluid Model (p. 727).

14.5.7.3. Solid-Solid Exchange Coefficient

The solid-solid exchange coefficient K_{ls} has the following form [634] (p. 1089) :

$$K_{ls} = \frac{3(1+e_{ls})\left(\frac{\pi}{2} + C_{fr,ls}\frac{\pi^2}{8}\right)\alpha_s \rho_s \alpha_l \rho_l (d_l + d_s)^2 g_{0,ls}}{2\pi\left(\rho_l d_l^3 + \rho_s d_s^3\right)}\left|\vec{v}_l - \vec{v}_s\right| \tag{14.292}$$

where

e_{ls} = the coefficient of restitution

$C_{fr,ls}$ = the coefficient of friction between the l^{th} and s^{th} solid-phase particles $\left(C_{fr,ls} = 0\right)$

d_l = the diameter of the particles of solid l

$g_{0,ls}$ = the radial distribution coefficient

Note that the coefficient of restitution is described in Solids Pressure (p. 694) and the radial distribution coefficient is described in Radial Distribution Function (p. 695).

14.5.7.4. Drag Modification

When using the Eulerian or Mixture multiphase models, Ansys Fluent can include a user-specified drag modification term in the calculation of interphase momentum exchange. This is introduced by replacing K_{pq} in Equation 14.204 (p. 664) with K'_{pq} where

$$K'_{pq} = \eta K_{pq} \tag{14.293}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

and K_{pq} is defined as in Equation 14.212 (p. 667). You can specify the modification factor, η, as a constant, a user-defined function, defined by the Brucato et al. correlation, or the near-wall drag modification.

14.5.7.4.1. Brucato et al. Correlation

The Brucato et al. correlation, [84] (p. 1057) , is appropriate for dilute gas-liquid flows [331] (p. 1072) and solid-liquid flows [448] (p. 1079) where the drag coefficient is increased by the liquid phase turbulence. For the Brucato et al. correlation, the drag modification factor is expressed as

$$\eta=(1+\gamma) \tag{14.294}$$

where

$$\gamma=K\left(\frac{d}{\lambda}\right)^3 \tag{14.295}$$

Here, $K=6.5\times10^{-6}$, d is the bubble diameter and λ is the Kolmogrov length scale given by:

$$\lambda=\left(\frac{\nu_l^3}{\varepsilon}\right)^{1/4} \tag{14.296}$$

where ν_l is the specific molecular viscosity of the liquid phase and ε is the average liquid phase turbulent eddy dissipation.

14.5.7.4.2. Near-Wall Drag Enhancement

The Ansys Fluent near-wall drag enhancement is effective for accurate predictions of the slip velocity in multiphase near-wall flows. The drag modification is well suited, for example, for air-water systems where a layer of equal-sized bubbles slides along the wall. In this scenario, the bubbles are in the flow region with the highest average strain ratio and highest turbulence, which can cause large bubbles to burst into smaller ones. The smaller bubbles can slide down the wall for a long period of time without breaking, only showing some deviation from an ideal spherical shape. When simulating such flows without the drag modification, the local accumulation of bubbles may lead to numerical problems.

The near-wall drag enhancement maintains a smooth transition from wall to bulk fluid, thus avoiding volume fraction stagnation and velocities overshoots. This improves the numerical robustness and the accuracy of the predicted near-wall velocity and void fraction values. It also decreases the instabilities and non-physical high velocity peaks close to the wall.

The enhancement formulation is proprietary to Ansys and therefore not published.

14.5.8. Lift Coefficient Correction

14.5.8.1. Shaver-Podowski Correction

The lift correction model by Shaver and Podowski [587] (p. 1087) is intended to damp the lift force near the wall, reducing it to zero when the gas is closer than one-bubble radius to the wall. The model improves the accuracy of the prediction of the void peak near the wall and overall robustness when used in conjunction with the turbulent dispersion and wall lubrication models.

The corrected lift coefficient is defined as:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$C_{CL}=\begin{cases} C_L & \frac{y_w}{d_b}\geq 1.0 \\ C_L\left(3\left(\frac{2y_w}{d_b}-1\right)^2-2\left(\frac{2y_w}{d_b}-1\right)^3\right) & 0.5\leq\frac{y_w}{d_b}<1.0 \\ 0 & \frac{y_w}{d_b}<0.5 \end{cases} \tag{14.297}$$

where C_L is the lift coefficient, y_w is the distance from the wall, and d_b is the bubble diameter.

14.5.9. Lift Force

For multiphase flows, Ansys Fluent can include the effect of lift forces on the secondary phase particles, droplets, or bubbles. These lift forces act on a particle mainly due to velocity gradients in the primary-phase flow field. The lift force will be more significant for larger particles, but the Ansys Fluent model assumes that the particle diameter is much smaller than the interparticle spacing. Thus, the inclusion of lift forces is not appropriate for closely packed particles or for very small particles.

From Drew (1993) [152] (p. 1061) , the lift force acting on a secondary phase p in a primary phase q can be calculated as:

$$\vec{F}_{lift}=-C_l\rho_q\alpha_p\left(\vec{v}_q-\vec{v}_p\right)\times\left(\nabla\times\vec{v}_q\right) \tag{14.298}$$

where

C_l = the lift coefficient (Lift Coefficient Models (p. 683))

ρ_q = the primary phase density

α_p = the secondary phase volume fraction

$\vec{v}_q$ = the primary phase velocity

$\vec{v}_p$ = the secondary phase velocity

The lift force $\vec{F}_{lift}$ will be added to the right-hand side of the momentum equation for both phases ($\vec{F}_{lift,q}=-\vec{F}_{lift,p}$).

In most cases, the lift force is insignificant compared to the drag force, so there is no reason to include this extra term. If the lift force is significant (for example, if the phases separate quickly), it may be appropriate to include this term. By default, $\vec{F}_{lift}$ is not included. The lift force and lift coefficient can be specified for each pair of phases, if desired.

Important:

It is important that if you include the lift force in your calculation, you need not include it everywhere in the computational domain since it is computationally expensive to converge. For example, in the wall boundary layer for turbulent bubbly flows in channels the lift force is significant when the slip velocity is large in the vicinity of high strain rates for the primary phase.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

14.5.9.1. Lift Coefficient Models

Fluent offers a variety of models for the lift coefficient, C_l, in Equation 14.298 (p. 682). These models are described in the following sections.

14.5.9.1.1. Moraga Lift Force Model

14.5.9.1.2. Saffman-Mei Lift Force Model

14.5.9.1.3. Legendre-Magnaudet Lift Force Model

14.5.9.1.4. Tomiyama Lift Force Model

14.5.9.1.5. Hessenkemper et al. Lift Force Model

Alternatively, you can specify a constant, or a User-Defined Function for the Lift Coefficient (`DEFINE_EXCHANGE_PROPERTY`): Refer to Including the Lift Force in the Fluent User's Guide for details about how to include lift force in your simulation.

14.5.9.1.1. Moraga Lift Force Model

The model developed by Moraga et al (1999) [455] (p. 1079) is applicable mainly to the lift force on spherical solid particles, though it can be applied to liquid drops and bubbles. In this model the lift coefficient combines opposing actions of two phenomena:

- classical aerodynamic lift resulting from the interaction between the dispersed phase particles and the primary phase shear
- the vorticity-induced lift resulting from interaction between particles and vortices shed by particle wakes

As a result, the lift coefficient is defined in terms of both particle Reynolds number and vorticity Reynolds number:

$$\mathrm{Re}_p=\frac{\rho_q|\vec{V}_q-\vec{V}_p|d_p}{\mu_q} \tag{14.299}$$

$$\mathrm{Re}_\omega=\frac{\rho_q|\nabla\times\vec{V}_q|d_p^2}{\mu_q} \tag{14.300}$$

Defining $\varphi=\mathrm{Re}_p\mathrm{Re}_\omega$, the Moraga lift coefficient is formulated as:

$$C_l=\begin{cases}0.0767 & \varphi\le 6000\\ -\left(0.12-0.2e^{-(\varphi/3.6)\times 10^{-5}}\right)e^{(\varphi/3)\times 10^{-7}} & 6000<\varphi<5\times 10^7\\ -0.6353 & \varphi\ge 5\times 10^7\end{cases} \tag{14.301}$$

14.5.9.1.2. Saffman-Mei Lift Force Model

The Saffman-Mei model is applicable mainly to spherical solid particles, though it can be applied to liquid drops that are not significantly distorted. As in the Moraga model (Moraga Lift Force Model (p. 683)), the lift coefficient is correlated in terms of particle Reynolds number and vorticity Reynolds number. According to Saffman (1965, 1968) [559] (p. 1085), [560] (p. 1085), for low Reynolds number flow past a spherical particle, the lift coefficient can be defined as:

$$\mathrm{C}_l=\frac{3}{2\pi\sqrt{\mathrm{Re}_\omega}}C_l' \tag{14.302}$$

where $C_l^{'}=6.46$ and $0 \le \mathrm{Re}_p \le \mathrm{Re}_\omega \le 1$.

Mei and Klausner (1994) [422] (p. 1077) extended the model to a higher range of particle Reynolds numbers. The Saffman-Mei model is empirically represented as:

$$C_l^{'}=\begin{cases} 6.46\times f\left(\mathrm{Re}_p,\mathrm{Re}_\omega\right) & \mathrm{Re}_p \le 40 \\ 6.46\times 0.0524\left(\beta \mathrm{Re}_p\right)^{1/2} & 40<\mathrm{Re}_p<100 \end{cases} \tag{14.303}$$

where

$$\beta=0.5\left(\mathrm{Re}_\omega/\mathrm{Re}_p\right)$$

$$f\left(\mathrm{Re}_p,\mathrm{Re}_\omega\right)=\left(1-0.3314\beta^{0.5}\right)e^{-0.1\mathrm{Re}_p}+0.3314\beta^{0.5}$$

14.5.9.1.3. Legendre-Magnaudet Lift Force Model

The Legendre-Magnaudet model (1998) [350] (p. 1073) is applicable primarily to small diameter spherical fluid particles, though it can be applied to non-distorted liquid drops and bubbles. In contrast to the Saffman-Mei model for rigid solid particles, the Legendre-Magnaudet model accounts for the momentum transfer between the flow around the particle and the inner recirculation flow inside the fluid particle caused by the fluid friction/stresses at the fluid interface. Therefore, the predicted lift force coefficients are approximately 2–5 times smaller than for rigid solid particles.

The range of validity given by Legendre and Magnaudet is as follows:

$$0.1<\mathrm{Re}_p<500 \quad , \quad Sr=2\beta \le 1$$

where β is defined as in the Saffman-Mei model (Saffman-Mei Lift Force Model (p. 683)). The lift force coefficient is then estimated as:

$$C_l=\sqrt{\left(C_{l,low\,\mathrm{Re}}\right)^2+\left(C_{l,high\,\mathrm{Re}}\right)^2} \tag{14.304}$$

where:

$$C_{l,\,low\,\mathrm{Re}}=\frac{6}{\pi^2}\left(\mathrm{Re}_p Sr\right)^{-0.5}\frac{2.55}{\left(1+0.2\frac{\mathrm{Re}_p}{Sr}\right)^{1.5}} \tag{14.305}$$

$$C_{l,high\,\mathrm{Re}}=\frac{1}{2}\frac{1+16\mathrm{Re}_p^{-1}}{1+29\mathrm{Re}_p^{-1}}$$

14.5.9.1.4. Tomiyama Lift Force Model

The Tomiyama model is applicable to the lift force on larger-scale deformable bubbles in the ellipsoidal and spherical cap regimes. As with the Tomiyama drag and wall lubrication models, this model depends on the Eötvös number. Its main feature is prediction of the cross-over point in bubble size at which particle distortion causes a reversal in the sign of the lift force. The model implemented in Fluent is a lightly modified version of the original Tomiyama lift model (1998) [653] (p. 1091) by Frank et al. (2004) [190] (p. 1063) :

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$C_l = \begin{cases} \min\left[0.288\tanh\left(0.121\mathrm{Re}_p\right), f\left(Eo'\right)\right] & Eo' \le 4 \\ f\left(Eo'\right) & 4 < Eo' \le 10 \\ -0.27 & 10 < Eo' \end{cases} \tag{14.306}$$

where

$$f\left(Eo'\right) = 0.00105Eo'^3 - 0.0159Eo'^2 - 0.0204Eo' + 0.474 \tag{14.307}$$

Eo' is a modified Eötvös number based on the long axis of the deformable bubble, d_h :

$$Eo' = \frac{g\left(\rho_q - \rho_p\right)d_h^2}{\sigma} \tag{14.308}$$

$$d_h = d_b\left(1 + 0.163Eo^{0.757}\right)^{1/3} \tag{14.309}$$

$$Eo = \frac{g\left(\rho_q - \rho_p\right)d_b^2}{\sigma} \tag{14.310}$$

where σ is the surface tension, g is gravity, and d_b is the bubble diameter.

14.5.9.1.5. Hessenkemper et al. Lift Force Model

The Hessenkemper et al. model [240] (p. 1067) is applicable to both small diameter spherical fluid particles and large-scale deformable fluid particles in the ellipsoidal and spherical cap regimes. It provides a better prediction of the lift coefficient sign and accounts for possible hydrodynamics effects produced by the presence of small surface-active impurities in the fluid. The model significantly improves estimation of both the transition point from a positive to negative lift sign (thus improving the overall void distribution in lift-driven air-water systems) and the lift coefficient for ellipsoidal bubbles. It also considers a correction by Ziegenhein and Tomiyama for deionized and tap water [240] (p. 1067), [731] (p. 1095).

The lift coefficient C_l is estimated by:

$$C_l = f\left(Sr, Re\right) - f\left(Eo'\right) \tag{14.311}$$

Here, $f\left(Sr, Re\right)$ is calculated similarly to Equation 14.305 (p. 684) in the Legendre-Magnaudet lift force model:

$$f\left(Sr, Re\right) = \left(\left(\frac{6}{\pi^2}\cdot\frac{1}{\left(Re\cdot Sr\right)^{1/2}}\cdot\frac{2.255}{\left(1+0.2\epsilon^{-2}\right)^{3/2}}\right)^2 + \left(0.5\frac{Re+16}{Re+29}\right)^2\right)^{1/2} \tag{14.312}$$

where:

- The Reynolds number Re is defined as:

$$Re = \frac{\rho_q\left|\vec{V_q} - \vec{V_p}\right|d_b}{\mu_q} \tag{14.313}$$

- The Strouhal number Sr is defined as:

$$Sr = \frac{d_b\gamma}{\left|\vec{V_q} - \vec{V_p}\right|} \tag{14.314}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

- ϵ is defined as:

$$\epsilon = \left(\frac{Sr}{Re} \right)^{1/2} \tag{14.315}$$

In the above equations,

ρ_q = primary phase density

V_q and V_p = primary and secondary phase velocities, respectively

d_b = bubble diameter

μ_q = viscosity of the primary phase

γ = shear stress

$f\left(Eo'\right)$ in Equation 14.311 (p. 685) is a correction for ellipsoidal bubbles expressed as:

$$f\left(Eo'\right) = \frac{\ln\left(1+\exp\left(-12\left(G\right)\right)\right)}{12} \tag{14.316}$$

with

$$G = 0.11\frac{\ln\left(1+\exp\left(4\left(Eo'-5.6\right)\right)\right)}{4} - 0.14\left(Eo'-5.2\right) - 0.44 \tag{14.317}$$

The horizontal Eötvös number Eo' is estimated as Equation 14.308 (p. 685) and is based on the horizontal diameter d_h calculated as:

$$d_h = d_B \sqrt[3]{1+AEo^B} \tag{14.318}$$

where Eo is the Eötvös number calculated by Equation 14.310 (p. 685).

In Equation 14.318 (p. 686), the coefficients A = 0.65 and B = 0.35 proposed by Ziegenhein and Lucas [730] (p. 1095) are used instead of those by Wellek et al. [696] (p. 1093) for a better prediction of the bubble major axis, and hence Eo'.

14.5.10. Wall Lubrication Force

For liquid-gas bubbly flows using the Eulerian model Ansys Fluent can include the effect of wall lubrication forces on the secondary phases (bubbles). The wall lubrication force tends to push the secondary phases away from walls. For example, in a bubbly upward flow in a vertical pipe this force results in the dispersed phase concentrating in a region near, but not immediately adjacent to, the wall.

The wall lubrication force acting on a secondary phase, p, in a primary phase, q, has the general form:

$$\vec{F}_{wl} = C_{wl}\rho_q\alpha_p \left| \left(\vec{v}_q - \vec{v}_p\right)_{\parallel} \right|^2 \vec{n}_w \tag{14.319}$$

where

C_{wl} = the wall lubrication coefficient (Wall Lubrication Models (p. 687))

ρ_q = the primary phase density

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

α_p = the secondary phase volume fraction

$\left|(\vec{v}_q - \vec{v}_p)_{\parallel}\right|$ = the phase relative velocity component tangential to the wall surface

$\vec{n}_w$ = the unit normal pointing away from the wall

14.5.10.1. Wall Lubrication Models

The wall lubrication models in Ansys Fluent differ in how they compute the wall lubrication coefficient, C_{wl} in Equation 14.319 (p. 686).

Ansys Fluent offers the following models:

14.5.10.1.1. Antal et al. Model

14.5.10.1.2. Tomiyama Model

14.5.10.1.3. Frank Model

14.5.10.1.4. Hosokawa Model

14.5.10.1.5. Lubchenko Model

You can also specify a user-defined wall lubrication coefficient with a User Defined Function using the DEFINE_EXCHANGE_PROPERTY macro. For details on using a UDF to define the wall lubrication coefficient, see `DEFINE_EXCHANGE_PROPERTY` in the Fluent Customization Manual.

14.5.10.1.1. Antal et al. Model

The model proposed by Antal et al. [20] (p. 1054) computes the coefficient, C_{wl} , as:

$$C_{wl} = \max\left(0, \frac{C_{w1}}{d_b} + \frac{C_{w2}}{y_w}\right) \tag{14.320}$$

where C_{w1}=-0.01 and C_{w2}=0.05 are non-dimensional coefficients, d_b is the bubble/particle diameter, and y_w is the distance to the nearest wall.

Note that C_{wl} is nonzero only within a thin layer adjacent to the wall that satisfies:

$$y_w \leq -\left(C_{w2}/C_{w1}\right) d_b \tag{14.321}$$

corresponding to $y_w \leq 5 d_b$ with the default values of C_{w1} and C_{w2}. As a result, the Antal model will only be active on a sufficiently fine mesh and grid independence can only be achieved with very fine meshes.

14.5.10.1.2. Tomiyama Model

The Tomiyama model [653] (p. 1091) modifies the wall lubrication force formulation of Antal [20] (p. 1054) based on the results of experiments with flow of air bubbles in glycerin in a pipe. The expression for C_{wl} for the Tomiyama model is:

$$C_{wl} = C_w \frac{d_b}{2}\left(\frac{1}{y_w^2} - \frac{1}{\left(D - y_w\right)^2}\right) \tag{14.322}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

In Equation 14.322 (p. 687), D is the pipe diameter and C_w depends on the Eotvos number, Eo. C_w is defined as

$$C_w = \begin{cases} 0.47 & Eo < 1 \\ e^{-0.933Eo+0.179} & 1 \le Eo \le 5 \\ 0.00599Eo - 0.0187 & 5 < Eo \le 33 \\ 0.179 & 33 \le Eo \end{cases} \tag{14.323}$$

where the Eotvos number, Eo is defined as

$$Eo = \frac{g(\rho_q - \rho_p)d_b^2}{\sigma}$$

and σ is the surface tensions coefficient. It has been noted by Frank et al. [190] (p. 1063) that, though the Tomiyama model has been found to be superior to the Antal model, it is restricted to flows in pipe geometries due to the dependence on pipe diameter in Equation 14.322 (p. 687).

14.5.10.1.3. Frank Model

The model of Frank et al. [190] (p. 1063), [191] (p. 1064) removes the dependence on pipe diameters in the Tomiyama model [653] (p. 1091). The Frank model defines the coefficient C_{wl} as

$$C_{wl} = C_w \max\left(0, \frac{1}{C_{wd}} \cdot \frac{1 - \frac{y_w}{C_{wc}d_b}}{y_w\left(\frac{y_w}{C_{wc}d_b}\right)^{m-1}}\right) \tag{14.324}$$

where

C_w is determined as a function of Eo as in Equation 14.323 (p. 688).

C_{wd} is the damping coefficient and determines the relative magnitude of the force.

C_{wc} is the cut-off coefficient and determines the distance to the wall within which the force is active.

m is the power law constant and is recommended to be between 1.5 and 2.

By default, C_{wd}=6.8, C_{wc}=10, and m=1.7.

14.5.10.1.4. Hosokawa Model

The model from Hosokawa et al. [257] (p. 1067) is based on experimental measurements that indicate that, for bubbles, the coefficient, C_w (Equation 14.323 (p. 688)) used in the Tomiyama and Frank models depends not only on the Eotvos number, but on the phase relative Reynolds number. The model proposes the following expression for C_w.

$$C_w = \max\left(\frac{7}{\mathrm{Re}_d^{1.9}}, 0.0217Eo\right) \tag{14.325}$$

where the phase relative Reynolds number, Re_d, is defined

$$\mathrm{Re}_d = \frac{\rho_q |\vec{v}_q - \vec{v}_p| d_b}{\mu_q} \tag{14.326}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

By default, the Hosokawa model computes C_{wl} as in Equation 14.324 (p. 688). Note that Equation 14.325 (p. 688) is only valid for liquid-gas bubble flows.

14.5.10.1.5. Lubchenko Model

The Lubchenko approach [392] (p. 1075) departs from treating wall lubrication as an artificial force exerted on bubbles near the wall and uses a more fundamental treatment, which is based on a geometric correction arising from the spherical shape of the bubbles. From this, a functional dependence for the gas volume fraction is derived, which is used to construct a wall lubrication force that regularized turbulent dispersion in the near-wall region. The model accounts for the decreasing cross-sectional area of bubbles.

If lift is considered, you should use this model together with the Shaver-Podowski lift correction (see Shaver-Podowski Correction (p. 681)), which will damp the lift force near the wall, thus allowing the turbulent dispersion to remain as the only interfacial force acting perpendicular to the wall.

The Lubcheko's derivation for the gradient of volume fraction term takes the following form [392] (p. 1075):

$$\nabla \alpha_p = \alpha_p \frac{1}{y_w} \frac{d_b - 2y_w}{d_b - y_w} \mathbf{n} \tag{14.327}$$

where

$\nabla \alpha_p$ = gradient of the volume fraction of the dispersed phase

α_p = volume fraction of the dispersed phase

y_w = distance from the wall

d_b = bubble diameter

$\mathbf{n}$ = wall normal vector

Substituting the gradient of the volume fraction term in the specific turbulent dispersion approach by Equation 14.327 (p. 689) yields the following equations for the wall lubrication model when using it together with the Burns et al. or the Lopez de Bertodano turbulent dispersion models:

- Burns et al. (Burns et al. Model (p. 691))

$$C_w = C_{TD} \frac{3}{4} \frac{C_D}{d_b} \left| \vec{v}_p - \vec{v}_q \right| \frac{\mu_L^{turb}}{\sigma_{TD}} \left(1 + \frac{\alpha_p}{1 - \alpha_p} \right) \nabla \alpha_p \tag{14.328}$$

- Lopez de Bertodano (Lopez de Bertodano Model (p. 691))

$$C_w = C_{TD} \rho_q k_q \nabla \alpha_p \tag{14.329}$$

In the above equations:

C_w = wall lubrication coefficient

C_{TD} = turbulent dispersion coefficient

C_D = drag coefficient

$\left| \vec{v}_p - \vec{v}_q \right|$ = slip velocity

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

μ_L^{turb} = turbulent viscosity

σ_{TD} = dispersion Prandtl number

ρ_q = density of the primary phase

k_q = turbulent dispersion of the primary phase

Equation 14.328 (p. 689) and Equation 14.329 (p. 689) are valid as long as $y_w < d_b/2$. If $y_w \geq d_b/2$, the Lubchenko wall lubrication force will not be used, and C_w will be equal to zero.

The Lubchenko wall lubrication model does not require any limiter or calibration coefficients in order to control its performance, thus significantly improving the generality, applicability, and ease of use of the approach.

14.5.11. Turbulent Dispersion Force

For multiphase turbulent flows using the Eulerian model, Ansys Fluent can include the effects of turbulent dispersion forces which account for the interphase turbulent momentum transfer. The turbulent dispersion force acts as a turbulent diffusion in dispersed flows. For example for a boiling flow in a heated vertical pipe, vapor is generated on heated wall surfaces. The turbulent dispersion force plays a crucial role in driving the vapor away from the vicinity of the wall towards the center of the pipe.

The turbulent dispersion force arises from averaging the interphase drag term. For a dispersed phase, p, and a continuous phase, q, the turbulent drag is modeled as:

$$K_{pq}(\tilde{\vec{v}}_p - \tilde{\vec{v}}_q) = K_{pq}(\vec{v}_p - \vec{v}_q) - K_{pq}\vec{v}_{dr} \tag{14.330}$$

The term on the left side is the instantaneous drag. The first term on the right hand side, $K_{pq}(\vec{v}_p - \vec{v}_q)$, appears in Equation 14.196 (p. 663) and represents the mean momentum exchange between the two phases. K_{pq} is the interphase exchange coefficient, described in Interphase Exchange Coefficients (p. 667), and $\vec{v}_p$ and $\vec{v}_q$ are the mean phase velocity vectors. The second term, $K_{pq}\vec{v}_{dr}$, is commonly referred to as the turbulent dispersion force:

$$\vec{F}_{td,q} = -\vec{F}_{td,p} = -f_{td,limiting} K_{pq}\vec{v}_{dr} \tag{14.331}$$

where $\vec{v}_{dr}$ is the drift velocity and accounts for the dispersion of the secondary phases due to transport by turbulent fluid motion. $f_{td,limiting}$ is a factor that can be used to impose a limiting function on the turbulent dispersion force. For clarity, the limiting factor is omitted from the descriptions of the turbulent dispersion models that follow. See Limiting Functions for the Turbulent Dispersion Force (p. 692) for details about the implementation of the limiting factor.

14.5.11.1. Models for Turbulent Dispersion Force

The following sections describe the models for turbulent dispersion force that are available in Ansys Fluent:

14.5.11.1.1. Lopez de Bertodano Model

14.5.11.1.2. Simonin Model

14.5.11.1.3. Burns et al. Model

14.5.11.1.4. Diffusion in VOF Model

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

You can also specify the turbulent dispersion force with a User Defined Function. For information on enabling turbulent dispersion force in your model, see Including the Turbulent Dispersion Force in the User's Guide.

14.5.11.1.1. Lopez de Bertodano Model

Instead of following Equation 14.331 (p. 690) and modeling the drift velocity, Lopez de Bertodano proposed the following formulation [386] (p. 1075):

$$\vec{F}_{td,q} = -\vec{F}_{td,p} = C_{TD}\rho_q k_q \nabla \alpha_p \tag{14.332}$$

where ρ_q is the continuous phase density, k_q is the turbulent kinetic energy in the continuous phase, and $\nabla \alpha_p$ is the gradient of dispersed phase volume fraction, and C_{TD} is a user-modifiable constant. By default, $C_{TD}=1$.

14.5.11.1.2. Simonin Model

Simonin and Viollet [595] (p. 1087) proposed that the drift velocity, $\vec{v}_{dr}$, is calculated from:

$$\vec{v}_{dr} = -\mathbf{D}_{pq} \cdot \left(\frac{\nabla \alpha_p}{\alpha_p} - \frac{\nabla \alpha_q}{\alpha_q} \right) \tag{14.333}$$

In Equation 14.333 (p. 691), $\mathbf{D}_{pq}$ is the fluid-particulate dispersion tensor.

For the dispersed two-equation turbulence model (k- ε Dispersed Turbulence Model (p. 707)), Ansys Fluent uses Tchen theory [246] (p. 1067) and assumes that $\mathbf{D}_{pq}$ is a scalar, expressed as given by $D_{t,pq}$ in Equation 14.436 (p. 714) with a dispersion Prandtl number, σ_{pq} ($\sigma_{pq}=0.75$ by default). The turbulent dispersion force is then given by:

$$\vec{F}_{td,q} = -\vec{F}_{td,p} = C_{TD} K_{pq} \frac{D_{t,pq}}{\sigma_{pq}} \left(\frac{\nabla \alpha_p}{\alpha_p} - \frac{\nabla \alpha_q}{\alpha_q} \right) \tag{14.334}$$

Where C_{TD} is a user-modifiable constant, which is set to 1 by default.

For the per-phase turbulence model, the turbulent dispersion force is given by:

$$\vec{F}_{td,q} = -\vec{F}_{td,p} = C_{TD} \frac{K_{pq}}{\sigma_{pq}} \left(\frac{\mu_p \nabla \alpha_p}{\rho_p \alpha_p} - \frac{\mu_q \nabla \alpha_q}{\rho_q \alpha_q} \right) \tag{14.335}$$

Where μ_p and ρ_p are the viscosity and density of the dispersed phase, respectively; and μ_q and ρ_q are the viscosity and density of the continuous phase, respectively.

For the mixture turbulence model (k- ε Mixture Turbulence Model (p. 705)), the dispersion scalar is equal to the mixture turbulent kinematic viscosity.

14.5.11.1.3. Burns et al. Model

Burns et al. [88] (p. 1058) derived a formulation based on Favre averaging of the drag term. The final expression is similar to Simonin's model. For the Burns model, the dispersion scalar is estimated by the turbulent viscosity of the continuous phase:

$$D_q = D_p = D_{tq} = \frac{\mu_{tq}}{\rho_q} \tag{14.336}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

and

$$\vec{F}_{td,q} = -\vec{F}_{td,p} = C_{TD} K_{pq} \frac{D_q}{\sigma_{pq}} \left(\frac{\nabla \alpha_p}{\alpha_p} - \frac{\nabla \alpha_q}{\alpha_q} \right) \tag{14.337}$$

By default, $C_{TD}=1$ and $\sigma_{pq}=0.9$. As in the Simonin model, when using the mixture turbulence model the dispersion scalar is equal to the mixture turbulent kinetic viscosity.

14.5.11.1.4. Diffusion in VOF Model

Instead of treating the turbulent dispersion as an interfacial momentum force in the phase momentum equations, you can model it as a turbulent diffusion term in the governing equations of phase volume fractions [612] (p. 1088). With the turbulent dispersion term, the governing equation for the volume fraction in phase q, Equation 14.193 (p. 662), becomes:

$$\frac{\partial}{\partial t}(\alpha_q \rho_q) + \nabla \cdot (\alpha_q \rho_q \vec{v}_q) = \nabla \left(\gamma_q \nabla \alpha_q \right) + \sum_{p=1}^{n} \left(\dot{m}_{pq} - \dot{m}_{qp} \right) + S_q \tag{14.338}$$

where γ_q is the diffusion coefficient in the q^{th} phase, and the term $\nabla \left(\gamma_q \nabla \alpha_q \right)$ is the turbulent dispersion term that must satisfy the constraint:

$$\sum_{q=1}^{n} \nabla \left(\gamma_q \nabla \alpha_q \right) = 0 \tag{14.339}$$

In order to satisfy Equation 14.339 (p. 692) the diffusion coefficients for the secondary phases are estimated from the phase turbulent viscosities, $\mu_{t,q}$ as

$$\gamma_q = \frac{\mu_{t,q}}{\sigma_q} \quad q = 2 \ldots n \tag{14.340}$$

where $\sigma_q=0.75$ by default.

For the primary phase, $q=1$, the diffusion term is

$$D_q = -\sum_{p=2}^{n} D_p \tag{14.341}$$

14.5.11.2. Limiting Functions for the Turbulent Dispersion Force

In some applications it is desirable to apply the turbulent dispersion force only in particular flow regimes or conditions. For example, a user may want to include the turbulent dispersion force only in a bubbly flow regime. To accommodate this, a limiting function can be applied through the factor, $f_{td,limiting}$, in Equation 14.331 (p. 690).

Ansys Fluent provides three options for determining $f_{td,limiting}$:

None

By default, no limiting is performed on the turbulent dispersion force ($f_{td,limiting}=1$).

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Standard

Ansys Fluent includes a standard limiting function that varies linearly from `0` to `1` based on the volume fraction of the dispersed phase(s), α_p:

$$f_{td,limiting}(\alpha_p)=\max\left(0,\min\left(1,\frac{\alpha_{p,2}-\alpha_p}{\alpha_{p,2}-\alpha_{p,1}}\right)\right) \tag{14.342}$$

By default,

$$\alpha_{p,1}=0.3$$
$$\alpha_{p,2}=0.7$$

If necessary, the values of $\alpha_{p,1}$ and $\alpha_{p,2}$ can be changed using the following `domainsetvar` scheme commands:

(domainsetvar <phase-id> 'mp/td/vof-lower-limit < $\alpha_{p,1}$ —value>)

(domainsetvar <phase-id> 'mp/td/vof-upper-limit < $\alpha_{p,2}$ —value>)

with appropriate values substituted for <phase-id> and < $\alpha_{p,i}$ —value>. Note that the value for <phase-id> for a given phase can be found from the **Phases** dialog box in the user interface.

User-Defined

You can specify your own limiting function by creating a User-Defined Function (UDF) using the `DEFINE_EXCHANGE_PROPERTY` macro (`DEFINE_EXCHANGE_PROPERTY`). Note that $f_{td,limiting}$ must have a value between `0` and `1`, and should vary continuously to ensure numerical stability and physical solutions.

14.5.12. Virtual Mass Force

For multiphase flows, you can optionally include the "virtual mass effect" that occurs when a secondary phase p accelerates relative to the primary phase q. The inertia of the primary-phase mass encountered by the accelerating particles (or droplets or bubbles) exerts a "virtual mass force" on the particles [152] (p. 1061). The virtual mass force is defined as:

$$\vec{F}_{vm}=C_{vm}\alpha_p\rho_q\left(\frac{d_q\vec{v}_q}{dt}-\frac{d_p\vec{v}_p}{dt}\right) \tag{14.343}$$

where C_{vm} is the virtual mass coefficient which typically has a value of `0.5`. The term $\frac{d_q}{dt}$ denotes the phase material time derivative of the form

$$\frac{d_q(\varphi)}{dt}=\frac{\partial(\varphi)}{\partial t}+(\vec{v}_q\cdot\nabla)\varphi \tag{14.344}$$

The virtual mass force $\vec{F}_{vm}$ will be added to the right-hand side of the momentum equation for both phases ($\vec{F}_{vm,q}=-\vec{F}_{vm,p}$).

The virtual mass effect is significant when the secondary phase density is much smaller than the primary phase density (for example, for a transient bubble column).

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

By default, the virtual mass is applied as an explicit source term. You can also choose to use an implicit approach which is recommended in steady-state coupled simulations where the explicit approach may fail to converge. Refer to Including the Virtual Mass Force for details on how to include virtual mass in your simulation.

14.5.13. Solids Pressure

For granular flows in the compressible regime (that is, where the solids volume fraction is less than its maximum allowed value), a solids pressure is calculated independently and used for the pressure gradient term, ∇p_s, in the granular-phase momentum equation. Because a Maxwellian velocity distribution is used for the particles, a granular temperature is introduced into the model, and appears in the expression for the solids pressure and viscosities. The solids pressure is composed of a kinetic term and a second term due to particle collisions:

$$p_s\alpha_s=\rho_s\Theta_s+2\rho_s\left(1+e_{ss}\right)\alpha_s^2 g_{0,ss}\Theta_s \tag{14.345}$$

where e_{ss} is the coefficient of restitution for particle collisions, $g_{0,ss}$ is the radial distribution function, and Θ_s is the granular temperature. Ansys Fluent uses a default value of 0.9 for e_{ss}, but the value can be adjusted to suit the particle type. The granular temperature Θ_s is proportional to the kinetic energy of the fluctuating particle motion, and will be described later in this section. The function $g_{0,ss}$ (described below in more detail) is a distribution function that governs the transition from the "compressible" condition with $\alpha < \alpha_{s,max}$, where the spacing between the solid particles can continue to decrease, to the "incompressible" condition with $\alpha=\alpha_{s,max}$, where no further decrease in the spacing can occur. A value of 0.63 is the default for $\alpha_{s,max}$, but you can modify it during the problem setup.

Other formulations that are also available in Ansys Fluent are [637] (p. 1090)

$$p_s=2\rho_s\left(1+e_{ss}\right)\alpha_s^2 g_{0,ss}\Theta_s \tag{14.346}$$

and [401] (p. 1076)

$$p_s=\alpha_s\rho_s\Theta_s\left[\left(1+4\alpha_s g_{0,ss}\right)+\frac{1}{2}\left[\left(1+e_{ss}\right)\left(1-e_{ss}+2\mu_{fric}\right)\right]\right] \tag{14.347}$$

where μ_{fric} is defined in [401] (p. 1076).

When more than one solids phase are calculated, the above expression does not take into account the effect of other phases. A derivation of the expressions from the Boltzman equations for a granular mixture are beyond the scope of this manual, however there is a need to provide a better formulation so that some properties may feel the presence of other phases. A known problem is that N solid phases with identical properties should be consistent when the same phases are described by a single solids phase. Equations derived empirically may not satisfy this property and need to be changed accordingly without deviating significantly from the original form. From [204] (p. 1064), a general solids pressure formulation in the presence of other phases could be of the form

$$p_q=\alpha_q\rho_q\Theta_q+\sum_{p=1}^{N}\frac{\pi}{3}g_{0,pq}d_{qp}^3 n_q n_p\left(1+e_{qp}\right)f\left(m_p,m_q,\Theta_p,\Theta_q\right) \tag{14.348}$$

where $d_{pq}=\frac{d_p+d_q}{2}$ is the average diameter, n_p, n_q are the number of particles, m_p and m_q are the masses of the particles in phases p and q, and f is a function of the masses of the particles and their

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

granular temperatures. For now, we have to simplify this expression so that it depends only on the granular temperature of phase q

$$p_q = \alpha_q \rho_q \Theta_q + \sum_{p=1}^{N} 2 \frac{d_{pq}^3}{d_q^3} \left(1+e_{pq}\right) g_{0,pq} \alpha_q \alpha_p \rho_q \Theta_q \tag{14.349}$$

Since all models need to be cast in the general form, it follows that

$$p_q = \alpha_q \rho_q \Theta_q + \left(\sum_{p=1}^{N} \frac{d_{pq}^3}{d_q^3} p_{c,qp} \right) \rho_q \Theta_q \tag{14.350}$$

where $p_{c,qp}$ is the collisional part of the pressure between phases q and p. In Equation 14.349 (p. 695), $p_{c,qp} = 2\left(1+e_{pq}\right) g_{0,pq} \alpha_q \alpha_p$.

The above expression reverts to the one solids phase expression when $N=1$ and $q=p$ but also has the property of feeling the presence of other phases.

14.5.13.1. Radial Distribution Function

The radial distribution function, g_0, is a correction factor that modifies the probability of collisions between grains when the solid granular phase becomes dense. This function may also be interpreted as the nondimensional distance between spheres:

$$g_0 = \frac{s+d_p}{s} \tag{14.351}$$

where s is the distance between grains. From Equation 14.351 (p. 695) it can be observed that for a dilute solid phase $s \to \infty$, and therefore $g_0 \to 1$. In the limit when the solid phase compacts, $s \to 0$ and $g_0 \to \infty$. The radial distribution function is closely connected to the factor χ of Chapman and Cowling's [99] (p. 1058) theory of nonuniform gases. χ is equal to 1 for a rare gas, and increases and tends to infinity when the molecules are so close together that motion is not possible.

In the literature there is no unique formulation for the radial distribution function. Ansys Fluent has a number of options:

- For one solids phase, use [204] (p. 1064) :

$$g_0 = \left[1 - \left(\frac{\alpha_s}{\alpha_{s,max}} \right)^{\frac{1}{3}} \right]^{-1} \tag{14.352}$$

  This is an empirical function and does not extend easily to n phases. For two identical phases with the property that $\alpha_s = \alpha_1 + \alpha_2$, the above function is not consistent for the calculation of the partial pressures p_1 and p_2, $p_s = p_1 + p_2$. In order to correct this problem, Ansys Fluent uses the following consistent formulation:

$$g_{0,ll} = \left[1 - \left(\frac{\alpha_s}{\alpha_{s,max}} \right)^{\frac{1}{3}} \right]^{-1} + \frac{1}{2} d_l \sum_{k=1}^{N} \frac{\alpha_k}{d_k} \tag{14.353}$$

  where $\alpha_{s,\max}$ is the packing limit, N is the total number of solid phases, d is the diameter of the phases, and

$$\alpha_s = \sum_{k=1}^{N} \alpha_k \tag{14.354}$$

and k are solid phases only.

- The following expression is also available [265] (p. 1068):

$$g_{0,ll} = \frac{1}{\left(1 - \frac{\alpha_s}{\alpha_{s,max}}\right)} + \frac{3}{2} d_l \sum_{k=1}^{N} \frac{\alpha_k}{d_k} \tag{14.355}$$

- Also available [401] (p. 1076), slightly modified for n solids phases, is the following:

$$g_{0,ll} = \frac{1 + 2.5\alpha_s + 4.59\alpha_s^2 + 4.52\alpha_s^3}{\left(1 - \left(\frac{\alpha_s}{\alpha_{s,max}}\right)^3\right)^{0.678}} + \frac{1}{2} d_l \sum_{k=1}^{N} \frac{\alpha_k}{d_k} \tag{14.356}$$

- The following equation [637] (p. 1090) is available:

$$g_{0,kl} = \frac{1}{\left(1 - \alpha_s\right)} + \frac{3\left(\sum_{k=1}^{N} \frac{\alpha_k}{d_k}\right)}{\left(1 - \alpha_s\right)^2 \left(d_j + d_k\right)} d_k d_l \tag{14.357}$$

When the number of solid phases is greater than 1, Equation 14.353 (p. 695) , Equation 14.355 (p. 696) and Equation 14.356 (p. 696) are extended to

$$g_{0,lm} = \frac{d_m g_{0,ll} + d_l g_{0,mm}}{d_m + d_l} \tag{14.358}$$

It is interesting to note that Equation 14.355 (p. 696) and Equation 14.356 (p. 696) compare well with [14] (p. 1053) experimental data, while Equation 14.357 (p. 696) reverts to the [93] (p. 1058) derivation.

14.5.14. Maximum Packing Limit in Binary Mixtures

The packing limit is not a fixed quantity and may change according to the number of particles present within a given volume and the diameter of the particles. Small particles accumulate in between larger particles increasing the packing limit. For a binary mixture Ansys Fluent uses the correlations proposed by [170] (p. 1062).

For a binary mixture with diameters $d_1 > d_2$, the mixture composition is defined as $X_1 = \frac{\alpha_1}{\alpha_1 + \alpha_2}$

where

$$X_1 <= \frac{\alpha_{1,max}}{\left(\alpha_{1,max} + \left(1 - \alpha_{1,max}\right)\alpha_{2,max}\right)} \tag{14.359}$$

as this is a condition for application of the maximum packing limit for binary mixtures.

The maximum packing limit for the mixture $\alpha_{s,max}$ is the minimum of the two expressions (Equation 14.360 (p. 696) and Equation 14.361 (p. 697):

$$\left(\alpha_{1,max} - \alpha_{2,max} + \left[1 - \sqrt{\frac{d_2}{d_1}}\right]\left(1 - \alpha_{1,max}\right)\alpha_{2,max}\right) * \left(\alpha_{1,max} + \left(1 - \alpha_{1,max}\right)\alpha_{2,max}\right)\frac{X_1}{\alpha_{1,max}} + \alpha_{2,max} \tag{14.360}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

and

$$\left[1-\sqrt{\frac{d_2}{d_1}}\right]\left(\alpha_{1,max}+\left(1-\alpha_{1,max}\right)\alpha_{2,max}\right)\left(1-X_1\right)+\alpha_{1,max} \tag{14.361}$$

The packing limit is used for the calculation of the radial distribution function.

For cases with more than two granular phases, the maximum packing limit (which is used in postprocessing calculations for the mixture phase) is computed as the maximum value of the **Packing Limit** for each granular phase.

14.5.15. Solids Shear Stresses

The solids stress tensor contains shear and bulk viscosities arising from particle momentum exchange due to translation and collision. A frictional component of viscosity can also be included to account for the viscous-plastic transition that occurs when particles of a solid phase reach the maximum solid volume fraction.

The collisional and kinetic parts, and the optional frictional part, are added to give the solids shear viscosity:

$$\mu_s=\mu_{s,col}+\mu_{s,kin}+\mu_{s,fr} \tag{14.362}$$

The third term in Equation 14.362 (p. 697) is only included when the solids fraction is greater than the friction packing limit.

14.5.15.1. Collisional Viscosity

The collisional part of the shear viscosity is modeled as [205] (p. 1064), [637] (p. 1090)

$$\mu_{s,col}=\frac{4}{5}\alpha_s\rho_s d_s g_{0,ss}\left(1+e_{ss}\right)\left(\frac{\Theta_s}{\pi}\right)^{1/2} \tag{14.363}$$

14.5.15.2. Kinetic Viscosity

Ansys Fluent provides two expressions for the kinetic part.

The default expression is from Syamlal et al. [637] (p. 1090) :

$$\mu_{s,kin}=\frac{d_s\rho_s\sqrt{\Theta_s\pi}}{6\left(3-e_{ss}\right)}\left[1+\frac{2}{5}\left(1+e_{ss}\right)\left(3e_{ss}-1\right)\alpha_s g_{0,ss}\right] \tag{14.364}$$

The following optional expression from Gidaspow et al. [205] (p. 1064) is also available:

$$\mu_{s,kin}=\frac{10\rho_s d_s\sqrt{\Theta_s\pi}}{96\alpha_s\left(1+e_{ss}\right)g_{0,ss}}\left[1+\frac{4}{5}g_{0,ss}\alpha_s\left(1+e_{ss}\right)\right]^2 \tag{14.365}$$

14.5.15.3. Bulk Viscosity

The solids bulk viscosity accounts for the resistance of the granular particles to compression and expansion. It has the following form from Lun et al. [394] (p. 1075):

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$\lambda_s = \frac{4}{3}\alpha_s \rho_s d_s g_{0,ss}(1+e_{ss})\left(\frac{\Theta_s}{\pi}\right)^{1/2} \tag{14.366}$$

Note that the bulk viscosity is set to a constant value of zero, by default. It is also possible to select the Lun et al. expression or use a user-defined function.

14.5.15.4. Frictional Viscosity

In dense flow at low shear, where the secondary volume fraction for a solid phase nears the packing limit, the generation of stress is mainly due to friction between particles. The solids shear viscosity computed by Ansys Fluent does not, by default, account for the friction between particles.

To model the frictional viscosity, Ansys Fluent provides the following models:

- Schaeffer [572] (p. 1086)
- constant
- user-defined

Schaeffer is given by:

$$\mu_{s,fr} = \frac{p_{friction}\sin\varphi}{2\sqrt{I_{2D}}} \tag{14.367}$$

where $p_{friction}$ is the frictional pressure, φ is the angle of internal friction, and I_{2D} is the second invariant of the deviatoric stress tensor. It is also possible to specify a constant or user-defined frictional viscosity.

In granular flows with high solids volume fraction, instantaneous collisions are less important. The application of kinetic theory to granular flows is no longer relevant since particles are in contact and the resulting frictional stresses need to be taken into account. Ansys Fluent extends the formulation of the frictional viscosity and employs other models, as well as providing new hooks for UDFs. See the Fluent Customization Manual for details.

The frictional stresses are usually written in Newtonian form:

$$\tau_{friction} = -p_{friction}\vec{\vec{I}} + \mu_{s,fr}\left(\nabla\vec{u}_s + \left(\nabla\vec{u}_s\right)^T\right) \tag{14.368}$$

The frictional stress is added to the stress predicted by the kinetic theory when the solids volume fraction exceeds a critical value. This critical value is referred to as the friction packing limit in Ansys Fluent. Then

$$p_S = p_{kinetic} + p_{friction} \tag{14.369}$$

$$\mu_S = \mu_{s,kin} + \mu_{s,fr} \tag{14.370}$$

The derivation of the frictional pressure is mainly semi-empirical, while the frictional viscosity can be derived from the first principles.

For frictional pressure, Ansys Fluent offers the following models:

- Johnson and Jackson [281] (p. 1069) model
- Syamlal et al [637] (p. 1090)
- based-ktgf

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

- user-defined

The Johnson and Jackson model for frictional pressure is defined as:

$$p_{friction} = Fr \frac{\left(\alpha_s - \alpha_{s,min}\right)^n}{\left(\alpha_{s,max} - \alpha_s\right)^p} \tag{14.371}$$

where Fr is a function of the volume fraction computed as:

$$Fr = 0.08\alpha_s \tag{14.372}$$

In Equation 14.371 (p. 699), $\alpha_{s,min}$ is the friction packing limit, $\alpha_{s,max}$ is the packing limit, the coefficient n = 2, and p = 5 [480] (p. 1081).

The second model that is employed is Syamlal et al. [637] (p. 1090). Comparing the two models results in the frictional normal stress differing by orders of magnitude.

The radial distribution function is an important parameter in the description of the solids pressure resulting from granular kinetic theory. If we use the models of Lun et al. [394] (p. 1075) or Gidaspow [204] (p. 1064) the radial function tends to infinity as the volume fraction tends to the packing limit. It would then be possible to use this pressure directly in the calculation of the frictional viscosity, as it has the desired effect. This approach is also available in Ansys Fluent as the **based-ktgf** option.

Important:

The introduction of the frictional viscosity helps in the description of frictional flows, however a complete description would require the introduction of more physics to capture the elastic regime with the calculation of the yield stress and the use of the flow-rule. These effects can be added by you via UDFs to model static regime. Small time steps are required to get good convergence behavior.

14.5.16. Granular Temperature

The granular temperature for the s^{th} solids phase is proportional to the kinetic energy of the particles' random motion. The formal expression is:

$$\Theta_s = \frac{1}{3} u_{s,i} u_{s,i} \tag{14.373}$$

In Equation 14.373 (p. 699) $u_{s,i}$ represents the i^{th} component of the fluctuating solids velocity in the Cartesian coordinate system. This is defined as an ensemble average of the particles' random velocity within a finite volume and time period. The averaging basis is particle number per unit volume following [99] (p. 1058) and [142] (p. 1061).

The transport equation derived from kinetic theory takes the form [142] (p. 1061):

$$\frac{3}{2}\left[\frac{\partial}{\partial t}\left(\rho_s \alpha_s \Theta_s\right) + \nabla \cdot \left(\rho_s \alpha_s \vec{v}_s \Theta_s\right)\right] = \left(-p_s \overline{\overline{I}} + \overline{\overline{\tau}}_s\right) : \nabla \vec{v}_s + \nabla \cdot \left(k_{\Theta_s} \nabla \Theta_s\right) - \gamma_{\Theta_s} + \varphi_{ls} \tag{14.374}$$

where

$\left(-p_s \overline{\overline{I}} + \overline{\overline{\tau}}_s\right) : \nabla \vec{v}_s$ = the generation of energy by the solid stress tensor

$k_{\Theta_s} \nabla \Theta_s$ = the diffusion of energy ($k_{\Theta s}$ is the diffusion coefficient)

γ_{Θ_s} = the collisional dissipation of energy

φ_{ls} = the energy exchange between the l^{th} fluid or solid phase and the s^{th} solid phase

Equation 14.374 (p. 699) contains the term $k_{\Theta_s}\nabla\Theta_s$ describing the diffusive flux of granular energy. When the default Syamlal et al. model [637] (p. 1090) is used, the diffusion coefficient for granular energy, k_{Θ_s} is given by

$$k_{\Theta_s}=\frac{15d_s\rho_s\alpha_s\sqrt{\Theta_s\pi}}{4(41-33\eta)}\left[1+\frac{12}{5}\eta^2(4\eta-3)\alpha_s g_{0,ss}+\frac{16}{15\pi}(41-33\eta)\eta\alpha_s g_{0,ss}\right] \tag{14.375}$$

where

$$\eta=\frac{1}{2}(1+e_{ss})$$

Ansys Fluent uses the following expression if the optional model of Gidaspow et al. [205] (p. 1064) is enabled:

$$k_{\Theta_s}=\frac{150\rho_s d_s\sqrt{(\Theta\pi)}}{384(1+e_{ss})g_{0,ss}}\left[1+\frac{6}{5}\alpha_s g_{0,ss}(1+e_s)\right]^2+2\rho_s\alpha_s{}^2 d_s(1+e_{ss})g_{0,ss}\sqrt{\frac{\Theta_s}{\pi}} \tag{14.376}$$

The collisional dissipation of energy, γ_{Θ_s}, represents the rate of energy dissipation within the s^{th} solids phase due to collisions between particles. This term is represented by the expression derived by Lun et al. [394] (p. 1075)

$$\gamma_{\Theta m}=\frac{12(1-e_{ss}^2)g_{0,ss}}{d_s\sqrt{\pi}}\rho_s\alpha_s^2\Theta_s^{3/2} \tag{14.377}$$

The transfer of the kinetic energy of random fluctuations in particle velocity from the s^{th} solids phase to the l^{th} fluid or solid phase is represented by φ_{ls} [205] (p. 1064) :

$$\varphi_{ls}=-3K_{ls}\Theta_s \tag{14.378}$$

Ansys Fluent allows you to solve for the granular temperature with the following options:

- algebraic formulation (the default)

  It is obtained by neglecting convection and diffusion in the transport equation, Equation 14.374 (p. 699) [637] (p. 1090).

- partial differential equation

  This is given by Equation 14.374 (p. 699) and it is allowed to choose different options for it properties.

- dpm-averaged granular temperature

  An alternative formulation available only with the Dense Discrete Phase Model (DDPM).

- constant granular temperature

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

This is useful in very dense situations where the random fluctuations are small.

- UDF for granular temperature

For a granular phase s, we may write the shear force at the wall in the following form:

$$\vec{\tau}_s = -\frac{\pi}{6}\sqrt{3}\,\varphi\frac{\alpha_s}{\alpha_{s,max}}\rho_s g_0\sqrt{\Theta_s}\,\vec{U}_{s,\parallel} \tag{14.379}$$

Here $\vec{U}_{s,\parallel}$ is the particle slip velocity parallel to the wall, φ is the specularity coefficient between the particle and the wall, $\alpha_{s,max}$ is the volume fraction for the particles at maximum packing, and g_0 is the radial distribution function that is model dependent.

The general boundary condition for granular temperature at the wall takes the form: [281] (p. 1069)

$$q_s = \frac{\pi}{6}\sqrt{3}\,\varphi\frac{\alpha_s}{\alpha_{s,max}}\rho_s g_0\sqrt{\Theta_s}\,\vec{U}_{s,\parallel}\cdot\vec{U}_{s,\parallel} - \frac{\pi}{4}\sqrt{3}\frac{\alpha_s}{\alpha_{s,max}}\left(1-e_{sw}^2\right)\rho_s g_0\Theta_s^{\frac{3}{2}} \tag{14.380}$$

14.5.17. Description of Heat Transfer

The internal energy balance for phase q is written in terms of the phase enthalpy, Equation 14.197 (p. 663) , defined by

$$H_q = \int c_{p,q}\,dT_q \tag{14.381}$$

where $c_{p,q}$ is the specific heat at constant pressure of phase q. The thermal boundary conditions used with multiphase flows are the same as those for a single-phase flow. See Cell Zone and Boundary Conditions in the User's Guide for details.

For dispersed-dispersed systems, Ansys Fluent applies the two-resistance model (Two-Resistance Model (p. 703)), internally using the same heat transfer coefficient method selected for the phase pair. For example, if you selected **ranz-marshall** correlation for the dispersed-dispersed system, then the resistance for each phase is calculated based on the Ranz-Marshall model.

14.5.17.1. The Heat Exchange Coefficient

The volumetric rate of energy transfer between phases, Q_{pq}, is assumed to be a function of the temperature difference and the interfacial area, A_i:

$$Q_{pq} = h_{pq}A_i\left(T_p - T_q\right) \tag{14.382}$$

where $h_{pq}\left(=h_{qp}\right)$ is the volumetric heat transfer coefficient between the p^{th} phase and the q^{th} phase. The heat transfer coefficient is related to the p^{th} phase Nusselt number, Nu_p, by

$$h_{pq} = \frac{\kappa_q Nu_p}{d_p} \tag{14.383}$$

Here κ_q is the thermal conductivity of the q^{th} phase, d_p is the bubble diameter.

When using the Eulerian multiphase, Ansys Fluent provides several methods for determining the volumetric heat transfer coefficient, h_{pq}, detailed in the following sections.

14.5.17.1.1. Constant

14.5.17.1.2. Nusselt Number
14.5.17.1.3. Ranz-Marshall Model
14.5.17.1.4. Tomiyama Model
14.5.17.1.5. Hughmark Model
14.5.17.1.6. Gunn Model
14.5.17.1.7. Two-Resistance Model
14.5.17.1.8. Fixed To Saturation Temperature
14.5.17.1.9. Constant Time Scale Method
14.5.17.1.10. User Defined

14.5.17.1.1. Constant

You can specify a constant value for the volumetric heat transfer coefficient, h_{pq}, by selecting the **constant-htc** method for heat transfer coefficient. If the Two-Resistance model is selected, the constant can be specified for each phase.

14.5.17.1.2. Nusselt Number

You can specify the Nusselt number, Nu_p, used in Equation 14.383 (p. 701) by selecting the **nusselt-number** method for heat transfer coefficient. For the Two-Resistance model, the Nusselt number can be specified for each phase. In such cases, the Nusselt number is always defined relative to the physical properties to which it pertains.

14.5.17.1.3. Ranz-Marshall Model

The correlation of Ranz and Marshall [539] (p. 1084), [540] (p. 1084) computes the Nusselt number as follows:

$$Nu_p = 2.0 + 0.6 \mathrm{Re}_p^{1/2} \mathrm{Pr}^{1/3} \tag{14.384}$$

where Re_p is the relative Reynolds number based on the diameter of the p^{th} phase and the relative velocity $\left|\vec{u_p} - \vec{u_q}\right|$, and Pr is the Prandtl number of the q^{th} phase:

$$Pr = \frac{c_{p_q} \mu_q}{\kappa_q} \tag{14.385}$$

14.5.17.1.4. Tomiyama Model

Tomiyama [653] (p. 1091) proposed a slightly different correlation for the interfacial heat transfer, applicable to turbulent bubbly flows with relatively low Reynolds number. For the Tomiyama model, the Nusselt number, Nu_p , is expressed:

$$Nu_p = 2.0 + 0.15 \mathrm{Re}_p^{0.8} \mathrm{Pr}^{0.5} \tag{14.386}$$

14.5.17.1.5. Hughmark Model

In order to extend the Ranz-Marshall model to a wider range of Reynolds numbers, Hughmark [261] (p. 1068) proposed the following correction:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$\mathrm{Nu_p}=\begin{cases}2+0.6\mathrm{Re_p}^{1/2}\mathrm{Pr}^{1/3} & 0\le \mathrm{Re_p}<776.06 \quad 0\le \mathrm{Pr}<250\\ 2+0.27\mathrm{Re_p}^{0.62}\mathrm{Pr}^{1/3} & 776.06\le \mathrm{Re_p} \quad 0\le \mathrm{Pr}<250\end{cases} \tag{14.387}$$

The Reynolds number crossover point is chosen to achieve continuity. The Hughmark model should not be used outside the recommended Prandtl number range

14.5.17.1.6. Gunn Model

In the case of granular flows (where $p=s$), you can also choose a Nusselt number correlation by Gunn [222] (p. 1065), applicable to a porosity range of 0.35–1.0 and a Reynolds number of up to 10^5:

$$Nu_s=\left(7-10\alpha_f+5\alpha_f^2\right)\left(1+0.7Re_s^{0.2}Pr^{1/3}\right)+\left(1.33-2.4\alpha_f+1.2\alpha_f^2\right)Re_s^{0.7}Pr^{1/3} \tag{14.388}$$

The Prandtl number is defined as above with $q=f$.

14.5.17.1.7. Two-Resistance Model

In some special situations, the use of an overall volumetric heat transfer coefficient is not sufficient to model the interphase heat transfer process accurately. A more general approach is to consider separate heat transfer processes with different heat transfer coefficients on either side of the phase interface. This generalization is referred to in Fluent as the two-resistance model.

At the interface between the q^{th} phase and the p^{th} phase, the temperature is assumed to be the same on both sides of the interface and is represented by T_s. Then the volumetric rates of phase heat exchange can be expressed as follows:

From the interface to the q^{th} phase:

$$Q_q=h_qA_i\left(T_s-T_q\right)+\dot{m}_{pq}H_{qs} \tag{14.389}$$

From the interface to the p^{th} phase:

$$Q_p=h_pA_i\left(T_s-T_p\right)-\dot{m}_{pq}H_{ps} \tag{14.390}$$

where h_q and h_p are the q^{th} and p^{th} phase heat transfer coefficients, and H_{qs} and H_{ps} are the q^{th} and p^{th} phase enthalpies, respectively.

Since neither heat nor mass can be stored on the phase interface, the overall heat balance must be satisfied:

$$Q_q+Q_p=0 \tag{14.391}$$

Therefore, in the absence of interphase mass transfer ($\dot{m}_{pq}=0$) the interfacial temperature is determined as follows:

$$T_s=\frac{h_qT_q+h_pT_p}{h_q+h_p} \tag{14.392}$$

and the interphase heat transfer is given by

$$Q_q=-Q_p=h_{pq}A_i\left(T_p-T_q\right), \quad \frac{1}{h_{pq}}=\frac{1}{h_q}+\frac{1}{h_p} \tag{14.393}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Hence, in the absence of interphase mass transfer the two-resistance model works somewhat similar to coupled wall thermal boundaries, wherein the interface temperature and the overall heat transfer coefficient are determined by the heat transfer rates on the two sides.

The phase heat transfer coefficients, h_q and h_p, can be computed using the same correlations that are available for computing the overall heat transfer coefficient, h_{pq}. In addition you can specify a zero-resistance condition on one side of the phase interface. This is equivalent to an infinite phase specific heat transfer coefficient. For example, if $h_q \to \infty$ its effect is to force the interface temperature to be the same as the phase temperature, $T_s = T_q$. You can also choose to use the *constant time scale return to saturation* method proposed by Lavieville et al. [343] (p. 1073) which is used by default for the interface to vapor heat transfer coefficient in the wall boiling models. Refer to Constant Time Scale Method (p. 704) for details of the formulation of the Lavieville et al method.

14.5.17.1.8. Fixed To Saturation Temperature

The **fixed-to-sat-temp** heat transfer model is intended to be used only when interphase mass transfer is being modeled.

In this model, the following conditions are assumed and are applied to Equation 14.389 (p. 703) , Equation 14.390 (p. 703) , and Equation 14.391 (p. 703) :

- All of the heat transferred to a phase-to-phase interface goes into mass transfer.
- The temperature at the **To-Phase** is equal to the saturation temperature.

The volumetric mass transfer rate, $\dot{m}_{pq}$, is determined from the selected mass transfer model (e.g. cavitation, evaporation-condensation Lee model, etc.). The energy relationships depending on the direction of mass transfer are:

- For $\dot{m}_{pq} \geq 0$ (mass transfers from the p^{th} phase to the q^{th} phase):

$$T_q = T_s \quad and \quad H_{qs} = H_{ps} = H_q(T_s) \tag{14.394}$$

- For $\dot{m}_{pq} < 0$ (mass transfers from the q^{th} phase to the p^{th} phase):

$$T_p = T_s \quad and \quad H_{qs} = H_{ps} = H_p(T_s) \tag{14.395}$$

14.5.17.1.9. Constant Time Scale Method

The interface to vapor heat transfer is calculated using the *constant time scale return to saturation* method [343] (p. 1073). It is assumed that the vapor retains the saturation temperature by rapid evaporation/condensation. The formulation is as follows:

$$\dot{q}_{vt} = \frac{\alpha_v \rho_v C_{p,v}}{\delta t} (T_{sat} - T_v) \tag{14.396}$$

Where δt is the time scale set to a default value of 0.05 and $C_{p,v}$ is the isobaric heat capacity. By default, Ansys Fluent uses this method in boiling applications to model the heat resistance on the vapor side when the heat transfer coefficient for the pair of boiling phases is modeled via any available formulation except the two-resistance model.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

14.5.17.1.10. User Defined

You can also specify the volumetric heat transfer coefficient, h_{pq} , as a user defined function using the DEFINE_EXCHANGE_PROPERTY UDF macro. See DEFINE_EXCHANGE_PROPERTY in the Fluent Customization Manual for details.

14.5.18. Turbulence Models

To describe the effects of turbulent fluctuations of velocities and scalar quantities in a single phase, Ansys Fluent uses various types of closure models, as described in Turbulence (p. 41). In comparison to single-phase flows, the number of terms to be modeled in the momentum equations in multiphase flows is large, and this makes the modeling of turbulence in multiphase simulations extremely complex.

Ansys Fluent provides three methods for modeling turbulence in multiphase flows within the context of the k - ε and k - ω models. In addition, Ansys Fluent provides two turbulence options within the context of the Reynolds stress models (RSM).

The k - ε and k - ω turbulence model options are:

- mixture turbulence model (the default)
- dispersed turbulence model
- turbulence model for each phase

Important:

Note that the descriptions of each method below are presented based on the standard k - ε model. The multiphase modifications to the k - ω , RNG and realizable k - ε models are similar, and are therefore not presented explicitly.

The RSM turbulence model options are:

- mixture turbulence model (the default)
- dispersed turbulence model

For either category, the choice of model depends on the importance of the secondary-phase turbulence in your application.

14.5.18.1. k-ε Turbulence Models

Ansys Fluent provides three turbulence model options in the context of the k - ε models: the mixture turbulence model (the default), the dispersed turbulence model, or a per-phase turbulence model.

14.5.18.1.1. k-ε Mixture Turbulence Model

The mixture turbulence model is the default multiphase turbulence model. It represents the first extension of the single-phase k - ε model, and it is applicable when phases separate, for stratified (or nearly stratified) multiphase flows, and when the density ratio between phases is close to 1.

In these cases, using mixture properties and mixture velocities is sufficient to capture important features of the turbulent flow.

The k and ε equations describing this model (and without including buoyancy, dilation, and source terms) are as follows:

$$\frac{\partial}{\partial t}(\rho_m k)+\nabla\cdot(\rho_m \vec{v}_m k)=\nabla\cdot\left(\left(\mu_m+\frac{\mu_{t,m}}{\sigma_k}\right)\nabla k\right)+G_{k,m}-\rho_m\epsilon+\Pi_{k_m} \tag{14.397}$$

and

$$\frac{\partial}{\partial t}\left(\rho_m \varepsilon\right)+\nabla\cdot\left(\rho_m \vec{v}_m \varepsilon\right)=\nabla\cdot\left(\left(\mu_m+\frac{\mu_{t,m}}{\sigma_\varepsilon}\right)\nabla\varepsilon\right)+\frac{\varepsilon}{k}\left(C_{1\varepsilon}G_{k,m}-C_{2\varepsilon}\rho_m\varepsilon\right)+\Pi_{\varepsilon_m} \tag{14.398}$$

where the mixture density, ρ_m, molecular viscosity, μ_m, and velocity, $\vec{v}_m$, are computed from

$$\rho_m=\sum_{i=1}^{N}\alpha_i\rho_i \tag{14.399}$$

$$\mu_m=\sum_{i=1}^{n}\alpha_i\mu_i \tag{14.400}$$

and

$$\vec{v}_m=\frac{\sum_{i=1}^{N}\alpha_i\rho_i\vec{v}_i}{\sum_{i=1}^{N}\alpha_i\rho_i} \tag{14.401}$$

where α_i, ρ_i, μ_i, and $\vec{v}_i$ are, respectively, the volume fraction, density, viscosity, and velocity of the ith phase.

The turbulent viscosity for the mixture, $\mu_{t,m}$, is computed from

$$\mu_{t,m}=\rho_m C_\mu\frac{k^2}{\varepsilon} \tag{14.402}$$

and the production of turbulence kinetic energy, $G_{k,m}$, is computed from

$$G_{k,m}=\mu_{t,m}\left(\nabla\vec{v}_m+\left(\nabla\vec{v}_m\right)^T\right):\nabla\vec{v}_m \tag{14.403}$$

The terms, Π_{k_m} and Π_{ε_m} are source terms that can be included to model the turbulent interaction between the dispersed phases and the continuous phase (Turbulence Interaction Models (p. 712)):

The turbulent viscosity for phase i is computed from

$$\mu_{t,i}=\frac{\rho_i}{\rho_m}\mu_{t,m} \tag{14.404}$$

The constants in these equations are the same as those described in Standard k-ε Model (p. 49) for the single-phase k - ε model.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

14.5.18.1.2. k- ε Dispersed Turbulence Model

The dispersed turbulence model is the appropriate model when the concentrations of the secondary phases are dilute, or when using the granular model. Fluctuating quantities of the secondary phases can be given in terms of the mean characteristics of the primary phase and the ratio of the particle relaxation time and eddy-particle interaction time.

The model is applicable when there is clearly one primary continuous phase and the rest are dispersed dilute secondary phases.

14.5.18.1.2.1. Assumptions

The dispersed method for modeling turbulence in Ansys Fluent assumes the following:

- a modified k - ε model for the continuous phase

  Turbulent predictions for the continuous phase are obtained using the standard k - ε model supplemented with extra terms that include the interphase turbulent momentum transfer.

- Tchen-theory correlations for the dispersed phases

  Predictions for turbulence quantities for the dispersed phases are obtained using the Tchen theory of dispersion of discrete particles by homogeneous turbulence [246] (p. 1067).

- interphase turbulent momentum transfer

  In turbulent multiphase flows, the momentum exchange terms contain the correlation between the instantaneous distribution of the dispersed phases and the turbulent fluid motion. It is possible to take into account the dispersion of the dispersed phases transported by the turbulent fluid motion.

- a phase-weighted averaging process

  The choice of averaging process has an impact on the modeling of dispersion in turbulent multiphase flows. A two-step averaging process leads to the appearance of fluctuations in the phase volume fractions. When the two-step averaging process is used with a phase-weighted average for the turbulence, however, turbulent fluctuations in the volume fractions do not appear. Ansys Fluent uses phase-weighted averaging, so no volume fraction fluctuations are introduced into the continuity equations.

14.5.18.1.2.2. Turbulence in the Continuous Phase

The eddy viscosity model is used to calculate averaged fluctuating quantities. The Reynolds stress tensor for continuous phase q takes the following form:

$$\overline{\overline{\tau}}_q'' = -\frac{2}{3}\left(\rho_q k_q + \rho_q \nu_{t,q} \nabla \cdot \vec{U}_q\right)\overline{\overline{I}} + \rho_q \nu_{t,q}\left(\nabla \vec{U}_q + \nabla \vec{U}_q^{\,T}\right) \tag{14.405}$$

where $\vec{U}_q$ is the phase-weighted velocity.

The turbulent viscosity $\mu_{t,q}$ is written in terms of the turbulent kinetic energy of phase q:

$$\mu_{t,q}=\rho_q C_\mu \frac{k_q^2}{\varepsilon_q} \tag{14.406}$$

and a characteristic time of the energetic turbulent eddies is defined as

$$\tau_{t,q}=\frac{3}{2} C_\mu \frac{k_q}{\varepsilon_q} \tag{14.407}$$

where ε_q is the dissipation rate and $C_\mu=0.09$.

The length scale of the turbulent eddies is

$$L_{t,q}=\sqrt{\frac{3}{2}}\, C_\mu \frac{k_q^{3/2}}{\varepsilon_q} \tag{14.408}$$

Turbulent predictions are obtained from the modified k - ε model. The transport equations (excluding buoyancy, dilation, and user-defined source terms) are:

$$\begin{aligned}\frac{\partial}{\partial t}(\alpha_q\rho_q k_q)+\nabla\cdot(\alpha_q\rho_q\vec{U}_q k_q) &= \nabla\cdot\left(\alpha_q\left(\mu_q+\frac{\mu_{t,q}}{\sigma_k}\right)\nabla k_q\right)+\alpha_q G_{k,q}\\ &\quad -\alpha_q\rho_q\epsilon_q+\alpha_q\rho_q\Pi_{k_q}\end{aligned} \tag{14.409}$$

and

$$\begin{aligned}\frac{\partial}{\partial t}\left(\alpha_q\rho_q\varepsilon_q\right)+\nabla\cdot\left(\alpha_q\rho_q\vec{U}_q\varepsilon_q\right) &= \nabla\cdot\left(\alpha_q\left(\mu_q+\frac{\mu_{t,q}}{\sigma_\varepsilon}\right)\nabla\varepsilon_q\right)\\ &\quad +\alpha_q\frac{\varepsilon_q}{k_q}\left(C_{1\varepsilon}G_{k,q}-C_{2\varepsilon}\rho_q\varepsilon_q\right)\\ &\quad +\alpha_q\rho_q\Pi_{\varepsilon_q}\end{aligned} \tag{14.410}$$

Here, the terms containing Π_{k_q} and Π_{ε_q} are source terms that can be included to model the influence of the dispersed phases on the continuous phase q (Turbulence Interaction Models (p. 712)): $G_{k,q}$ is the production of turbulent kinetic energy, as defined in Modeling Turbulent Production in the k-ε Models (p. 57). All other terms have the same meaning as in the single-phase k - ε model.

14.5.18.1.2.3. Turbulence in the Dispersed Phase

The turbulence quantities for the dispersed phase are not obtained from transport equations. Time and length scales that characterize the motion are used to evaluate dispersion coefficients, correlation functions, and the turbulent kinetic energy of each dispersed phase.

14.5.18.1.3. k-ε Turbulence Model for Each Phase

The most general multiphase turbulence model solves a set of k and ε transport equations for each phase. This turbulence model is the appropriate choice when the turbulence transfer among the phases plays a dominant role.

Note that, since Ansys Fluent is solving two additional transport equations for each secondary phase, the per-phase turbulence model is more computationally intensive than the dispersed turbulence model.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

14.5.18.1.3.1. Transport Equations

The Reynolds stress tensor and turbulent viscosity are computed using Equation 14.405 (p. 707) and Equation 14.406 (p. 708). Turbulence predictions are obtained from

$$\begin{aligned}\frac{\partial}{\partial t}\left(\alpha_q\rho_q k_q\right)+\nabla\cdot\left(\alpha_q\rho_q\vec{U}_q k_q\right)= &\ \nabla\cdot\left(\alpha_q\left(\mu_q+\frac{\mu_{t,q}}{\sigma_k}\right)\nabla k_q\right)\\ &+\left(\alpha_q G_{k,q}-\alpha_q\rho_q\varepsilon_q\right)\\ &+\sum_{l=1}^{N}K_{lq}\left(C_{lq}k_l-C_{ql}k_q\right)\\ &-\sum_{l=1}^{N}K_{lq}\left(\vec{U}_l-\vec{U}_q\right)\cdot\frac{\mu_{t,l}}{\alpha_l\rho_l\sigma_l}\nabla\alpha_l\\ &+\sum_{l=1}^{N}K_{lq}\left(\vec{U}_l-\vec{U}_q\right)\cdot\frac{\mu_{t,q}}{\alpha_q\rho_q\sigma_q}\nabla\alpha_q\\ &+\Pi_{k_q}\end{aligned} \tag{14.411}$$

and

$$\begin{aligned}\frac{\partial}{\partial t}\left(\alpha_q\rho_q\varepsilon_q\right)+\nabla\cdot\left(\alpha_q\rho_q\vec{U}_q\varepsilon_q\right)= &\ \nabla\cdot\left(\alpha_q\left(\mu_q+\frac{\mu_{t,q}}{\sigma_\varepsilon}\right)\nabla\varepsilon_q\right)\\ &+\frac{\varepsilon_q}{k_q}\left[C_{1\varepsilon}\alpha_q G_{k,q}-C_{2\varepsilon}\alpha_q\rho_q\varepsilon_q\right.\\ &+C_{3\varepsilon}\left(\sum_{l=1}^{N}K_{lq}\left(C_{lq}k_l-C_{ql}k_q\right)\right.\\ &-\sum_{l=1}^{N}K_{lq}\left(\vec{U}_l-\vec{U}_q\right)\cdot\frac{\mu_{t,l}}{\alpha_l\rho_l\sigma_l}\nabla\alpha_l\\ &\left.\left.+\sum_{l=1}^{N}K_{lq}\left(\vec{U}_l-\vec{U}_q\right)\cdot\frac{\mu_{t,q}}{\alpha_q\rho_q\sigma_q}\nabla\alpha_q\right)\right]\\ &+\Pi_{\varepsilon_q}\end{aligned} \tag{14.412}$$

The terms C_{lq} and C_{ql} can be approximated as

$$C_{lq}=2, C_{ql}=2\left(\frac{\eta_{lq}}{1+\eta_{lq}}\right) \tag{14.413}$$

where η_{lq} is defined by Equation 14.435 (p. 714). The terms Π_{k_q} and Π_{ε_q} are source terms which can be included to model the influence of the inter-phase turbulence interaction (Turbulence Interaction Models (p. 712)).

14.5.18.2. RSM Turbulence Models

Multiphase turbulence modeling typically involves two equation models that are based on single-phase models and often cannot accurately capture the underlying flow physics. Additional turbulence modeling for multiphase flows is diminished even more when the basic underlying single-phase model cannot capture the complex physics of the flow. In such situations, the logical next step is to combine the Reynolds stress model with the multiphase algorithm in order to handle challenging

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

situations in which both factors, RSM for turbulence and the Eulerian multiphase formulation, are a precondition for accurate predictions [115] (p. 1059).

The phase-averaged continuity and momentum equations for a continuous phase are:

$$\frac{\partial}{\partial t}\left(\overline{\alpha_c}\rho_c\right)+\nabla\cdot\left(\overline{\alpha_c}\rho_c\tilde{U}_c\right)=0 \tag{14.414}$$

$$\frac{\partial}{\partial t}\left(\overline{\alpha_c}\rho_{rmc}\tilde{U}_c\right)+\nabla\cdot\left(\overline{\alpha_c}\rho_{rmc}\tilde{U}_c\otimes\tilde{U}_c\right)=-\overline{\alpha_c}\nabla\tilde{p}+\nabla\cdot\tilde{\tau}_c^t+F_{Dc} \tag{14.415}$$

For simplicity, the laminar stress-strain tensor and other body forces such as gravity have been omitted from Equation 14.414 (p. 710) - Equation 14.415 (p. 710). The tilde denotes phase-averaged variables while an overbar (for example, $\overline{\alpha_c}$ reflects time-averaged values. In general, any variable Φ can have a phase-average value defined as

$$\tilde{\Phi}_c=\frac{\overline{\alpha_c\Phi_c}}{\overline{\alpha_c}} \tag{14.416}$$

Considering only two phases for simplicity, the drag force between the continuous and the dispersed phases can be defined as:

$$F_{Dc}=K_{dc}\left[\left(\tilde{U}_d-\tilde{U}_c\right)-\left(\frac{\overline{\alpha_d u_d'}}{\overline{\alpha_d}}-\frac{\overline{\alpha_c u_c'}}{\overline{\alpha_c}}\right)\right] \tag{14.417}$$

where K_{dc} is the drag coefficient. Several terms in the Equation 14.417 (p. 710) need to be modeled in order to close the phase-averaged momentum equations. Full descriptions of all modeling assumptions can be found in [114] (p. 1059). This section only describes the different modeling definition of the turbulent stresses $\tilde{\tau}^t$ that appears in Equation 14.415 (p. 710).

The turbulent stress that appears in the momentum equations need to be defined on a per-phase basis and can be calculated as:

$$\tilde{\tau^t}_k=-\overline{\alpha_k}\rho_k\tilde{R}_{k,ij} \tag{14.418}$$

where the subscript k is replaced by c for the primary (that is, continuous) phase or by d for any secondary (that is, dispersed) phases. As is the case for single-phase flows, the current multiphase Reynolds stress model (RSM) also solves the transport equations for Reynolds stresses R_{ij}. Ansys Fluent includes two methods for modeling turbulence in multiphase flows within the context of the RSM model: the dispersed turbulence model, and the mixture turbulence model.

14.5.18.2.1. RSM Dispersed Turbulence Model

The dispersed turbulence model is used when the concentrations of the secondary phase are dilute and the primary phase turbulence is regarded as the dominant process. Consequently, the transport equations for turbulence quantities are only solved for the primary (continuous) phase, while the predictions of turbulence quantities for dispersed phases are obtained using the Tchen theory. The transport equation for the primary phase Reynolds stresses in the case of the dispersed model are:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$\frac{\partial}{\partial t}\left(\overline{\alpha}\rho\widetilde{R}_{ij}\right)+\frac{\partial}{\partial x_k}\left(\overline{\alpha}\rho\widetilde{U}_k\widetilde{R}_{ij}\right) = -\overline{\alpha}\rho\left(\widetilde{R}_{ik}\frac{\partial\widetilde{U}_j}{\partial x_k}+\widetilde{R}_{jk}\frac{\partial\widetilde{U}_i}{\partial x_k}\right)+\frac{\partial}{\partial x_k}\left[\overline{\alpha}\mu\ \frac{\partial}{\partial x_k}\left(\widetilde{R}_{ij}\right)\right]$$
$$-\frac{\partial}{\partial x_k}\left[\overline{\alpha}\rho\overline{u_i'u_j'u_k'}+\overline{\alpha}p'\left(\delta_{jk}u_i'+\delta_{ik}u_j'\right)\right]+\overline{\alpha}\overline{p'\left(\frac{\partial u_i'}{\partial x_j}+\frac{\partial u_j'}{\partial x_i}\right)}$$
$$-\overline{\alpha}\rho\widetilde{\epsilon}_{ij}-2\overline{\alpha}\rho\Omega_k\left(\widetilde{R}_{jm}\epsilon_{ikm}+\widetilde{R}_{im}\epsilon_{jkm}\right)+\Pi_{R,ij} \tag{14.419}$$

The variables in Equation 14.419 (p. 711) are per continuous phase c and the subscript is omitted for clarity. In general, the terms in Equation 14.419 (p. 711) are modeled in the same way as for the single phase case described in Reynolds Stress Model (RSM) (p. 96). The last term, $\Pi_{R,ij}$, takes into account the interaction between the continuous and the dispersed phase turbulence. A general model for this term can be of the form:

$$\Pi_{R,ij}=K_{dc}C_{1,dc}\left(R_{dc,ij}-R_{c,ij}\right)+K_{dc}C_{2,dc}a_{dc,i}b_{dc,j} \tag{14.420}$$

where C_1 and C_2 are unknown coefficients, $a_{dc,i}$ is the relative velocity, $b_{dc,j}$ represents the drift or the relative velocity, and $R_{dc,ij}$ is the unknown particulate-fluid velocity correlation. To simplify this unknown term, the following assumption has been made:

$$\Pi_{R,ij}=\frac{2}{3}\delta_{ij}\Pi_k \tag{14.421}$$

where δ_{ij} is the Kronecker delta, and Π_k represents the modified version of the original Simonin model [596] (p. 1087).

$$\Pi_{kc}=K_{dc}\left(\widetilde{k}_{dc}-2\widetilde{k}_c+\widetilde{V}_{rel}\cdot\widetilde{V}_{drift}\right) \tag{14.422}$$

where $\widetilde{K}_c$ represents the turbulent kinetic energy of the continuous phase, $\widetilde{k}_{dc}$ is the continuous-dispersed phase velocity covariance and finally, $\widetilde{V}_{rel}$ and $\widetilde{V}_{drift}$ stand for the relative and the drift velocities, respectively. In order to achieve full closure, the transport equation for the turbulent kinetic energy dissipation rate ($\tilde{\varepsilon}$) is required. The modeling of $\tilde{\varepsilon}$ together with all other unknown terms in Equation 14.422 (p. 711) are modeled in the same way as in [114] (p. 1059).

14.5.18.2.2. RSM Mixture Turbulence Model

The main assumption for the mixture model is that all phases share the same turbulence field which consequently means that the term Π_R in the Reynolds stress transport equations (Equation 14.419 (p. 711)) is neglected. Apart from that, the equations maintain the same form but with phase properties and phase velocities being replaced with mixture properties and mixture velocities. The mixture density, for example, can be expressed as

$$\rho_m=\sum_{i=1}^{N}\overline{\alpha}_i\rho_i \tag{14.423}$$

while mixture velocities can be expressed as

$$\widetilde{U}_m=\frac{\sum_{i=1}^{N}\overline{\alpha}_i\rho_i\widetilde{U}_i}{\sum_{i=1}^{N}\overline{\alpha}_i\rho_i} \tag{14.424}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

where N is the number of species.

14.5.18.3. Turbulence Interaction Models

When using a turbulence model in an Eulerian multiphase simulation, Fluent can optionally include the influence of the dispersed phase on the multiphase turbulence equations. The influence of the dispersed phases is represented by source terms (Π_{k_q} and Π_{ε_q} in Equation 14.409 (p. 708) and Equation 14.410 (p. 708)) whose form will depend on the model chosen.

You can choose from the following models for turbulence interaction.

14.5.18.3.1. Simonin et al.

14.5.18.3.2. Troshko-Hassan

14.5.18.3.3. Sato

14.5.18.3.4. None

Turbulence interaction can be included with any of the multiphase turbulence models and formulations in Fluent.

Note:

For the ω-based turbulence models, the source term will take the form:

$$\Pi_{\omega q}=\frac{\Pi_{\varepsilon q}}{C_\mu k}-\frac{\omega \Pi_{kq}}{k} \tag{14.425}$$

See Including Turbulence Interaction Source Terms in the *Fluent User's Guide* for details about how to include turbulence interaction in your simulation.

14.5.18.3.1. Simonin et al.

In the Simonin et al. model [596] (p. 1087), the turbulence interaction is modeled by additional source terms in the turbulence transport equation(s). The Simonin model is only available with Dispersed and Per Phase turbulence models.

14.5.18.3.1.1. Formulation in Dispersed Turbulence Models

14.5.18.3.1.1.1. Continuous Phase

The term Π_{k_q} is derived from the instantaneous equation of the continuous phase and takes the following form, where M represents the number of secondary phases:

$$\Pi_{k_q}=C_s\sum_{p=1}^{M}\frac{K_{pq}}{\alpha_q\rho_q}X_{pq}\left(<\vec{v}_q''\cdot\vec{v}_p''>+\left(\vec{U}_p-\vec{U}_q\right)\cdot\vec{v}_{dr}\right) \tag{14.426}$$

which can be simplified to

$$\Pi_{k_q}=C_s\sum_{p=1}^{M}\frac{K_{pq}}{\alpha_q\rho_q}X_{pq}\left(k_{pq}-2k_q+\vec{v}_{pq}\cdot\vec{v}_{dr}\right) \tag{14.427}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

where: C_s is a user-modifiable model constant. By default, C_s=1. k_{pq} is the covariance of the velocities of the continuous phase q and the dispersed phase p, calculated from Equation 14.436 (p. 714). $\vec{v}_{pq}$ is the relative velocity. $\vec{v}_{dr}$ is the drift velocity, calculated from Equation 14.333 (p. 691).

$X_{pq}=\frac{\rho_p}{\rho_p+C_{VM}\rho_q}$ For granular flows, $X_{pq}\sim 1$.

Equation 14.427 (p. 712) can be split into two terms as follows:

$$\Pi_{k_q}=C_s\sum_{p=1}^{M}\frac{K_{pq}}{\alpha_q\rho_q}X_{pq}(k_{pq}-2k_q)+C_s\sum_{p=1}^{M}\frac{K_{pq}}{\alpha_q\rho_q}X_{pq}\vec{v}_{pq}\cdot\vec{v}_{dr} \tag{14.428}$$

The second term in Equation 14.428 (p. 713) is related to drift velocity and is referred to as the drift turbulent source. It is an option term in the turbulent kinetic energy source and can be included as described in Including Turbulence Interaction Source Terms in the *Fluent User's Guide*.

Π_{ε_q} is modeled according to Elgobashi et al. [163] (p. 1062) :

$$\Pi_{\varepsilon_q}=C_{3\varepsilon}\frac{\varepsilon_q}{k_q}\Pi_{k_q} \tag{14.429}$$

where $C_{3\varepsilon}$=1.2.

14.5.18.3.1.1.2. Dispersed Phases

For the dispersed phases, the characteristic particle relaxation time connected with inertial effects acting on a dispersed phase p is defined as:

$$\tau_{pq}^{F}=\frac{\rho_q d_p^2}{18\mu_q f\left(\mathrm{Re},\alpha_q\right)}\left(1+C_{VM}\frac{\rho_q}{\rho_p}\right) \tag{14.430}$$

where $f\left(\mathrm{Re},\alpha_q\right)$ is the drag function described in Interphase Exchange Coefficients (p. 667).

The time scale of the energetic turbulent eddies is defined as:

$$\tau_q^t=\frac{3}{2}C_\mu\frac{k_q}{\varepsilon_q} \tag{14.431}$$

The eddy particle interaction time is mainly affected by the crossing-trajectory effect [125] (p. 1060), and is defined as

$$\tau_{pq}^t=\frac{\tau_q^t}{\sigma_1\sqrt{(1+C_\beta\xi_\tau^2)}} \tag{14.432}$$

where σ_1=1, the parameter, ξ_τ, is given by

$$\xi_\tau=\frac{\left|\vec{U}_p-\vec{U}_q\right|}{\sqrt{\frac{2}{3}k_q}} \tag{14.433}$$

and

$$C_\beta = 1.8 - 1.35\cos^2\theta \tag{14.434}$$

where θ is the angle between the mean particle velocity and the mean relative velocity. The ratio between these two characteristic times is written as

$$\eta_{pq} = \frac{\tau_{pq}^t}{\tau_{pq}^F} \tag{14.435}$$

Following Simonin [596] (p. 1087), Ansys Fluent writes the turbulence quantities for dispersed phase p as follows:

$$\begin{aligned} k_p &= k_q\left(\frac{b^2+\eta_{pq}}{1+\eta_{pq}}\right) \\ k_{pq} &= 2k_q\left(\frac{b+\eta_{pq}}{1+\eta_{pq}}\right) \\ D_{t,pq} &= \frac{1}{3}k_{pq}\tau_{pq}^t \\ D_p &= D_{t,pq} + \left(\frac{2}{3}k_p - b\frac{1}{3}k_{pq}\right)\tau_{pq}^F \\ b &= \left(1+C_{VM}\right)\left(\frac{\rho_p}{\rho_q}+1\right)^{-1} \end{aligned} \tag{14.436}$$

and $C_{\mathrm{VM}} = 0.5$ is the added-mass coefficient. For granular flows, b is negligible. The viscosity for the secondary phase, p for granular flows is approximated as:

$$\nu_p = \frac{1}{3}k_{pq}\tau_{pq}^t + \frac{2}{3}k_p\tau_{pq}^F \tag{14.437}$$

while for bubbly flows it can be left as $D_{t,pq}$.

14.5.18.3.1.2. Formulation in Per Phase Turbulence Models

For Per Phase turbulence models, only the second term from Equation 14.428 (p. 713) (the drift turbulence source) is added in the phase turbulence modeling equations.

For the turbulent kinetic energy equations:

Continuous Phase

$$\Pi_{k_q} = C_s\alpha_q\sum_{p=1}^{M}K_{pq}\vec{v}_{pq}.\vec{v}_{dr} \tag{14.438}$$

Dispersed Phases

$$\Pi_{k_p} = C_s\alpha_p K_{pq}\vec{v}_{pq}.\vec{v}_{dr} \quad p = 1...m \tag{14.439}$$

Turbulence dissipation sources for all phases are computed as:

$$\Pi_{\varepsilon_r} = C_{3\varepsilon}\frac{\varepsilon_k}{k_r}\Pi_{k_r} \quad r = 0...m \tag{14.440}$$

14.5.18.3.2. Troshko-Hassan

Troshko and Hassan [656] (p. 1091) proposed an alternative model to account for the turbulence of the dispersed phase in the k - ε equations.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

14.5.18.3.2.1. Troshko-Hassan Formulation in Mixture Turbulence Models

In the Mixture turbulence models, the Troshko-Hassan turbulence interaction terms are:

$$\Pi_{k_m}=C_{ke}\sum_{p=1}^{M}K_{pq}\left|\vec{U}_p-\vec{U}_q\right|^2 \tag{14.441}$$

$$\Pi_{\varepsilon_m}=C_{td}\frac{1}{\tau_p}\Pi_{k_m} \tag{14.442}$$

By default, $C_{ke}=0.75$ and $C_{td}=0.45$. These values are user-modifiable as described in Including Turbulence Interaction Source Terms in the *Fluent User's Guide*. τ_p is the characteristic time of the induced turbulence defined as

$$\tau_p=\frac{2C_{VM}d_p}{3C_D\left|\vec{U}_p-\vec{U}_q\right|} \tag{14.443}$$

C_{VM} is the virtual mass coefficient and C_D is the drag coefficient.

14.5.18.3.2.2. Troshko-Hassan Formulation in Dispersed Turbulence Models

14.5.18.3.2.2.1. Continuous Phase

In the Dispersed turbulence models, the term Π_{k_q} is calculated as follows:

$$\Pi_{k_q}=C_{ke}\sum_{p=1}^{M}\frac{K_{pq}}{\alpha_q\rho_q}\left|\vec{U}_p-\vec{U}_q\right|^2 \tag{14.444}$$

and Π_{ε_q} is calculated as

$$\Pi_{\varepsilon_q}=C_{td}\frac{1}{\tau_p}\Pi_{k_q} \tag{14.445}$$

By default, $C_{ke}=0.75$ and $C_{td}=0.45$. These values are user-modifiable as described in Including Turbulence Interaction Source Terms in the *Fluent User's Guide*. τ_p is the characteristic time of the induced turbulence defined as in Equation 14.443 (p. 715).

14.5.18.3.2.2.2. Dispersed Phases

For the Troshko-Hassan model, the kinematic turbulent viscosity of the dispersed phase is calculated as

$$\nu_p=\nu_q \tag{14.446}$$

14.5.18.3.2.3. Troshko-Hassan Formulation in Per-Phase Turbulence Models

14.5.18.3.2.3.1. Continuous Phase

In the Per-Phase turbulence models, the continuous phase equations are modified with the following terms:

$$\Pi_{k_q}=C_{ke}\alpha_q\sum_{p=1}^{M}K_{pq}\left|\vec{U}_p-\vec{U}_q\right|^2 \tag{14.447}$$

$$\Pi_{\varepsilon_q}=C_{td}\frac{1}{\tau_p}\Pi_{k_q} \tag{14.448}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

By default, C_{ke}=0.75 and C_{td}=0.45. These values are user-modifiable as described in Including Turbulence Interaction Source Terms in the *Fluent User's Guide*. τ_p is the characteristic time of the induced turbulence defined as in Equation 14.443 (p. 715).

14.5.18.3.2.3.2. Dispersed Phases

The dispersed phase equations are also modified with source terms defined as follows:

$$\Pi_{k_p}=C_{ke}\alpha_p K_{pq}\left|\vec{U}_p-\vec{U}_q\right|^2 \quad p=1...M \tag{14.449}$$

$$\Pi_{\varepsilon_p}=C_{td}\frac{1}{\tau_p}\Pi_{k_p} \quad p=1...M \tag{14.450}$$

14.5.18.3.3. Sato

Unlike the Simonin and Troshko-Hassan models, the Sato model [568] (p. 1086) does not add explicit source terms to the turbulence equations. Instead, in an attempt to incorporate the effect of the random primary phase motion induced by the dispersed phase in bubbly flow, Sato et al. proposed the following relation:

$$\nu_q=C_\mu\frac{k^2}{\varepsilon}+C_{\mu,p}\alpha_p d_p\left|\vec{U}_p-\vec{U}_q\right| \tag{14.451}$$

where the relative velocity and the diameter of the dispersed phase represent velocity and time scales and $C_{\mu,p}$=0.6.

For the mixture model

$$C_\mu k^2/\varepsilon=\mu_{tq}/\rho_q$$

where μ_{tq} is the primary phase turbulent viscosity before the Sato correction, which is calculated from Equation 14.404 (p. 706).

For the dispersed and per-phase turbulence models, k and ε are the primary phase turbulence intensity and eddy dissipation rate, respectively.

14.5.18.3.4. None

If None is selected, then no source terms are added to account for turbulent interaction. This is appropriate if you prefer to add your own source terms using the `DEFINE_SOURCE` UDF macro (`DEFINE_SOURCE` in the *Fluent Customization Manual*).

14.5.19. Solution Method in Ansys Fluent

For Eulerian multiphase calculations, Ansys Fluent can solve the phase momentum equations, the shared pressure, and phasic volume fraction equations in a coupled and segregated fashion. The coupled solution for multiphase flows is discussed in detail in Coupled Solution for Eulerian Multiphase Flows in the User's Guide. When solving the equations in a segregated manner, Ansys Fluent uses the phase coupled SIMPLE (PC-SIMPLE) algorithm [665] (p. 1091) for the pressure-velocity coupling. PC-SIMPLE is an extension of the SIMPLE algorithm [504] (p. 1082) to multiphase flows. The velocities are solved coupled by phases, but in a segregated fashion. The block algebraic multigrid scheme used by the density-based solver described in [694] (p. 1093) is used to solve a vector equation formed by the velocity components of all phases simultaneously. Then, a pressure correction equation is built

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

based on total volume continuity rather than mass continuity. Pressure and velocities are then corrected so as to satisfy the continuity constraint.

14.5.19.1. The Pressure-Correction Equation

For incompressible multiphase flow, the pressure-correction equation takes the form:

$$\sum_{k=1}^{n}\frac{1}{\rho_{rk}}\left\{\frac{\partial}{\partial t}\alpha_k\rho_k+\nabla\cdot\alpha_k\rho_k\vec{v}_k^{'}+\nabla\cdot\alpha_k\rho_k\vec{v}_k^{*}-\left(\sum_{l=1}^{n}\left(\dot{m}_{lk}-\dot{m}_{kl}\right)\right)\right\}=0 \tag{14.452}$$

where ρ_{rk} is the phase reference density for the k^{th} phase (defined as the total volume average density of phase k), $\vec{v}_k^{'}$ is the velocity correction for the k^{th} phase, and $\vec{v}_k^{*}$ is the value of $\vec{v}_k$ at the current iteration. The velocity corrections are themselves expressed as functions of the pressure corrections.

14.5.19.2. Volume Fractions

The volume fractions are obtained from the phase continuity equations. In discretized form, the equation of the k^{th} volume fraction is

$$a_{p,k}\alpha_k=\sum_{nb}\left(a_{nb,k}\alpha_{nb,k}\right)+b_k=R_k \tag{14.453}$$

In order to satisfy the condition that all the volume fractions sum to one,

$$\sum_{k=1}^{n}\alpha_k=1 \tag{14.454}$$

14.5.20. Algebraic Interfacial Area Density (AIAD) Model

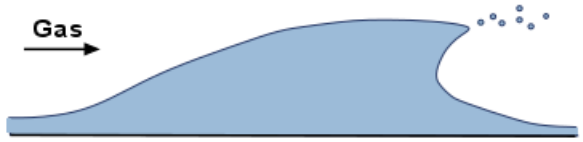
Higher turbulent gas-liquid bubbly flows are encountered in many industrial applications, including petrochemical, pharmaceutical, biochemical, nuclear, and metallurgical industries [449] (p. 1079). One common example of such flows is annular two-phase flow that occurs in boilers, heat exchangers, natural gas well, and so on. In the annular flow pattern, gas flows at high velocity through the core of the pipe, and the liquid film flows at lower velocity around the pipe wall. The high gas velocity results in large shear velocities, which, in turn, lead to high interfacial shear stress. The liquid film interface becomes unstable, and the droplets are torn from the interface and entrained in the gas core. Droplet entrainment changes flow characteristics. Different factors affect the entrained liquid fraction and the circumferential distribution of liquid film thickness on the wall. Gravity causes the liquid film drainage. Evaporation can also deplete the liquid wall film resulting in dry patches near the top of the pipe. These effects become more significant with increasing gas velocity and can result in equipment damage.

The traditional multiphase approaches to solving such problems do not predict the correct transitional flow behavior and mechanics behind droplet entrainment. In such regimes, the liquid droplet entrainment has a significant impact on the mechanisms for mass, momentum, and energy transfer. Hydrodynamic and surface forces can cause a significant deformation of the liquid film interface, resulting in the breakage of the continuous film surface into smaller dispersed droplets. The deformation depends on the flow pattern and the interface shape.

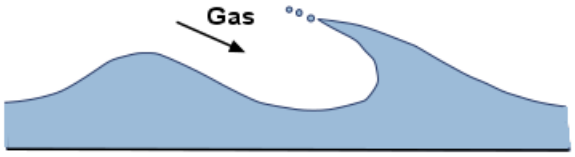
Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Ishii and Grolmes ([270] (p. 1068)) identified that liquid droplets can be entrained into a gas flow in the following patterns:

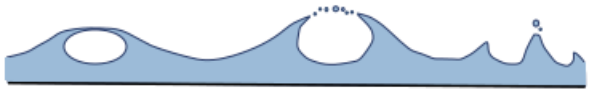
- Tearing off the tops of the crests of large-amplitude roll waves by the turbulent gas flow. The drag force acting on the wave top deforms the interface against the retaining force of the liquid surface tension.



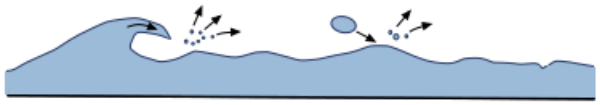
- Undercutting the liquid film by the gas flow. The gas penetrates the wave which starts to bulge and may eventually break.



- Bursting of gas bubbles. The thinning liquid film of the bubble rising to the gas-liquid interface rupture into droplets of various sizes.



- Impinging of a wave or droplet onto the film interface. A large amplitude wave may collapse into the liquid body generating droplets. Also, some droplet created by the previous mechanisms may impinge into the liquid phase producing smaller droplets.



Ansys Fluent uses the Algebraic Interfacial Area Density (AIAD) approach that offers the universal droplet entrainment model that covers all of the entrainment patterns and predicts the rate of entrainment. The AIAD approach models the momentum exchange dependent on the morphological form of the flow pattern. The model uses the volume fraction values to differentiate between bubbles, droplets, and the interface.

The AIAD framework was originally developed by Thomas Höhne from the Helmholtz-Zentrum Dresden-Rossendorf in close cooperation with Ansys ([252] (p. 1067), [249] (p. 1067), [253] (p. 1067), [162] (p. 1062), [135] (p. 1060), [449] (p. 1079), [452] (p. 1079)) to overcome known limitations of the modeling of separated / stratified flows. Using the volume fraction compression scheme, which is part of the Multi-Fluid VOF model, the AIAD model supports the simulation of three phases such as continuous gas, continuous liquid, and a dispersed entrained phase (that is, droplets). Depending of the detected morphology (dispersed droplets, dispersed bubbles, or free surface), different drag and interfacial area density

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

formulations are used. The AIAD mass transfer mechanism is used to model the breakage of the continuous liquid phase into the dispersed phase of the same material (entrainment), and inclusion of the dispersed phase into the continuous fluid phase (absorption). The entrained phase can be further extended via population balance approaches (direct quadrature method of moments (DQMOM), inhomogeneous discrete method, and so on), through which the physical distribution of the entrained phase can be obtained.

The correlations for the interfacial area density and the drag coefficient cover a full range of phase volume fractions from gas-only to liquid-only. The AIAD approach improves the physical modeling in the asymptotic limit between bubbly and droplet flows by using blending functions based on volume fraction. These functions for droplets, bubbles, and free surface morphologies (f_D, f_B, and f_{FS}) are defined as:

$$f_D = \left[1+e^{a_D(\alpha_L-\alpha_{D,limit})}\right]^{-1} \tag{14.455}$$

$$f_B = \left[1+e^{a_B(\alpha_G-\alpha_{B,limit})}\right]^{-1} \tag{14.456}$$

$$f_{FS} = 1-f_D-f_B \tag{14.457}$$

where α_L and α_G are volume fractions of the liquid and gaseous phases, respectively, a_D and a_B are the blending coefficients for droplets and bubbles, respectively, $\alpha_{D,limit}$ and $\alpha_{B,limit}$ are the volume fraction limiters for droplets and bubbles, respectively, and the subscripts D, B, and FS refer to the droplets, bubbles, and free surface, respectively. The default values are $a_D = a_B = 50$. These values are based on a number of parametric studies ([252] (p. 1067), [249] (p. 1067), [253] (p. 1067)). The default value for $\alpha_{D,limit}$ and $\alpha_{B,limit}$ is 0.3, which is chosen based on the approximated critical volume fraction.

14.5.20.1. Modeling Interfacial Area

The bubbles and droplets that leave the free-surface interface are initially assumed to be spherical with constant diameters d_B and d_D, respectively. The interfacial area densities for these phases $A_{i,D}$ and $A_{i,B}$ are calculated as:

$$A_{i,D} = \frac{6\alpha_L}{d_D} \tag{14.458}$$

$$A_{i,B} = \frac{6\alpha_D}{d_B} \tag{14.459}$$

For the free surface interphase, the interfacial area density A_{FS} depends on the gradient of the liquid volume fraction α_L:

$$A_{i,FS} = |\nabla \alpha_L| = \frac{\partial \alpha_L}{\partial n} \tag{14.460}$$

where n is the normal vector of the free surface.

The sum of the interfacial areas $A_{i,j}$ weighted by the corresponding blending function f_j gives the local interfacial area A_i :

$$A_i = f_{FS} A_{i,FS} + f_B A_{i,B} + f_D A_{i,D} \tag{14.461}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

14.5.20.2. Modeling Free-Surface Drag

Unlike one-velocity methods, such as Volume of Fluid (VOF), multi-fluid approaches consider velocity and turbulence of each phase, which induces the velocity difference between the fluids (slip velocity). The drag between the phases can be expressed using the volumetric density and the area density instead of the surface area:

$$\left|F_D\right| = C_D A_i \frac{1}{2} \rho \left|U\right|^2 \tag{14.462}$$

where C_D is the drag coefficient, U is the slip velocity, and ρ is the mixture density. If the free surface flow is considered, the phase averaged density is used:

$$\rho = \alpha_G \rho_G + \alpha_L \rho_L \tag{14.463}$$

where ρ_G and ρ_L are the densities of the gas and liquid phases, respectively.

In free surface flows, Equation 14.462 (p. 720) and Equation 14.463 (p. 720) do not represent realistic behavior of the physics at the interface. Höhne and Valle ([252] (p. 1067)) assume that the shear stress close to the free surface acts similarly to wall shear stress on both sides of the interface to reduce the velocity differences on both phases. The free-surface interface region behaves as a wall with a wall-like shear stress that acts at the free surface, which can lead to a decrease of gas velocity. The gradients of the void fraction determine the normal vector components. The shear stress at the interface is calculated by computing the scalar product between the gradient normal to the interface and the gradients of the two fluid velocities.

The wall-like free surface shear stress vector can be calculated as a product of the viscous stress tensor and the surface normal vector $\bar{n}$:

$$\tau_w = \begin{bmatrix} \tau_{xx} & \tau_{xy} & \tau_{xz} \\ \tau_{yx} & \tau_{yy} & \tau_{yz} \\ \tau_{zx} & \tau_{zy} & \tau_{zz} \end{bmatrix} \cdot \begin{bmatrix} n_x \\ n_y \\ n_z \end{bmatrix} \tag{14.464}$$

which translates into:

$$\begin{aligned} \tau_{w,x} &= \tau_{xx} n_x + \tau_{xy} n_y + \tau_{xz} n_z \\ \tau_{w,y} &= \tau_{yx} n_x + \tau_{yy} n_y + \tau_{yz} n_z \\ \tau_{w,z} &= \tau_{zx} n_x + \tau_{zy} n_y + \tau_{zz} n_z \end{aligned} \tag{14.465}$$

Equation 14.465 (p. 720) can be rearranged as:

$$\begin{aligned} \tau_{w,x} &= \left[n_x \mu \left(2\frac{\partial u}{\partial x} \right) \right] + \left[n_y \mu \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) \right] + \left[n_z \mu \left(\frac{\partial u}{\partial z} + \frac{\partial w}{\partial x} \right) \right] \\ \tau_{w,y} &= \left[n_x \mu \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) \right] + \left[n_y \mu \left(2\frac{\partial v}{\partial y} \right) \right] + \left[n_z \mu \left(\frac{\partial u}{\partial z} + \frac{\partial w}{\partial u} \right) \right] \\ \tau_{w,z} &= \left[n_x \mu \left(\frac{\partial w}{\partial x} + \frac{\partial u}{\partial z} \right) \right] + \left[n_y \mu \left(\frac{\partial w}{\partial y} + \frac{\partial v}{\partial z} \right) \right] + \left[n_z \mu \left(2\frac{\partial w}{\partial z} \right) \right] \end{aligned} \tag{14.466}$$

$$\tau_w = \sqrt{\tau_{w,x}^2 + \tau_{w,y}^2 + \tau_{w,z}^2} \tag{14.467}$$

where μ is the dynamic viscosity. The drag-force correlation (Equation 14.462 (p. 720)) can be the presented as the wall shear stress force acting at the free surface:

$$F_w = \tau_i A = F_D \tag{14.468}$$

The modification of the drag coefficient for the free surface can be written as:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$C_{D,FS}=\frac{2\left(\alpha_L\tau_{W,L}+\alpha_G\tau_{W,G}\right)}{\rho\left|U\right|^2} \tag{14.469}$$

where $\tau_{W,L}$ and $\tau_{W,G}$ are the wall-like shear stresses.

The AIAD model also uses the definition of drag coefficients for the bubble and droplet phases $C_{D,B}$ and $C_{D,D}$. The default values of 0.44 for these coefficients corresponds to a typical spherical fluid particle flow. You can define your own coefficients using user-defined functions (UDFs).

The total drag coefficient is calculated as:

$$C_D = f_{FS}\,C_{D,FS} + f_B\,C_{D,B} + f_D\,C_{D,D} \tag{14.470}$$

14.5.20.3. Modeling Sub-grid Wave Turbulence Contribution (SWT)

In traditional two-phase flow problems, the small waves at the interface created by Kelvin-Helmholtz instabilities are neglected. However, although these waves are smaller than the mesh size, they may have a significant influence on the turbulence kinetic energy of the liquid phase.

Gravity and surface tension tend to stabilize the liquid surface [80] (p. 1057). The surface behavior depends on the following two parameters:

- Turbulence Froude number: Is the ratio of fluids inertia to gravitational forces:

  $$Fr=\frac{q}{(2gL)^{1/2}} \tag{14.471}$$

  where q is the turbulent velocity [m/s], g is gravity [m/s^2], and L is the length scale [m].

- Weber number: Is a measure of the relative importance of the fluid's inertia compared to its surface tension:

  $$We=\frac{q^2L\rho_L}{2\sigma} \tag{14.472}$$

  where σ is the surface tension coefficient, and ρ_L is the density of the liquid phase.

These two numbers should be considered to correctly model the behavior of the interface. The AIAD takes these numbers into account by delineating a critical region of parameter space between quiescent surfaces and surfaces that completely break up.

Based on the Froude and Weber numbers, the following regimes can be distinguished ([80] (p. 1057), [250] (p. 1067)):

- $Fr << 1$, $We << 1$: Weak turbulence

  The turbulence is not strong enough to cause significant surface disturbance.

- $Fr >> 1$, $We << 1$: Knobbly flow

  The turbulence is sufficient to deform the surface against gravity, but only at small length scales. Surface tension creates a very smooth and rounded interface surface.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

- $Fr << 1, We >> 1$: Turbulence dominated by gravity

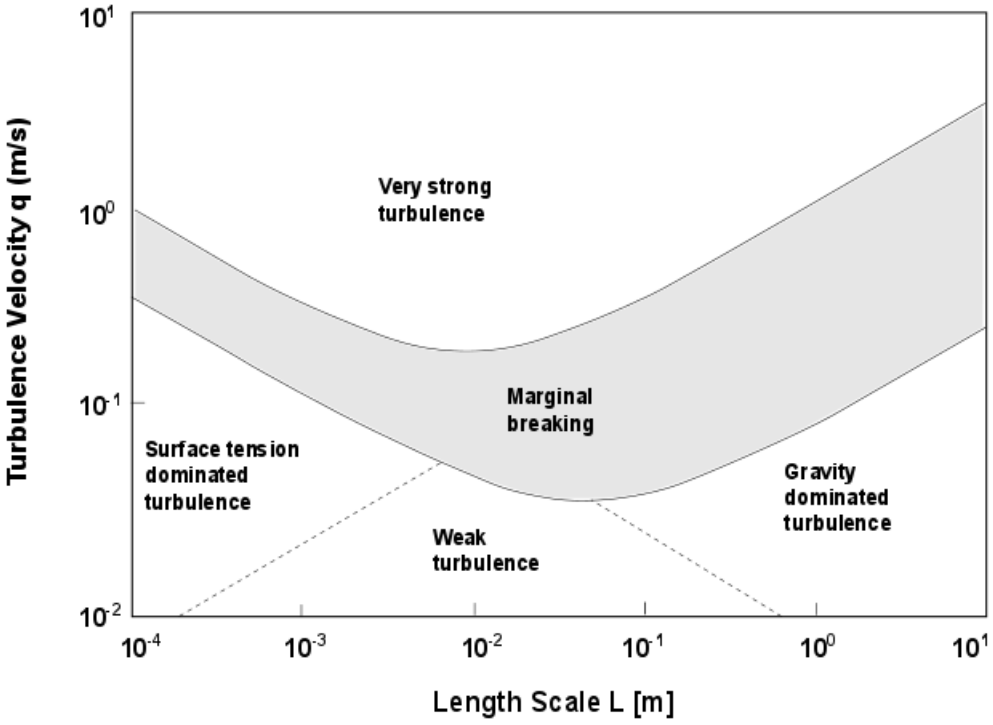
  Surface deformation is countered by gravity, resulting in a nearly flat interphase surface. The turbulent energy is high enough to disturb the surface at relatively small scales, creating small regions of waves, vortex dimples, and scars. This is the most common state in nature.

- $Fr >> 1, We >> 1$: Strong turbulence

  The turbulence is strong enough to counter gravity, and the surface tension is no longer sufficient to keep the flow stable.

Figure 14.11: Length-Turbulence Velocity Diagram for Water (p. 722) depicts the length-velocity diagram for water [80] (p. 1057). It shows a prediction of the interface shape in the logarithmic space of the length scale L and the liquid turbulence $q=\sqrt{2k}$.

Figure 14.11: Length-Turbulence Velocity Diagram for Water



The shaded area represents the region of marginal breaking estimated by the critical Weber number and the critical Froude number. This region also shows the variations in the interface which is no longer smooth due to the turbulent flow. Because of the multi-scale nature of turbulence, these regions may occur side-by-side. The production term for the turbulent kinetic energy caused by these disturbances can be expressed as:

$$P_{k,\,SWT}=f_{FS}\frac{2}{3}\frac{\partial U_i}{\partial x_i}\rho_L k_{sw} \tag{14.473}$$

The contribution to the turbulent kinetic energy produced by these unresolved waves can be calculated by the following:

$$k_{sw}=0.5\left(q_u^2-q_l^2\right) \tag{14.474}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Lower and upper bounds of the region that represents the changes between an interface disturbed by turbulences and an interface that reaches a breaking point are defined as ([250] (p. 1067)):

$$q_l^2 \approx \left(\frac{5}{3}-\frac{\pi}{2}\right)\frac{g_n L}{125}+\frac{(\pi-2)\sigma}{5L} \tag{14.475}$$

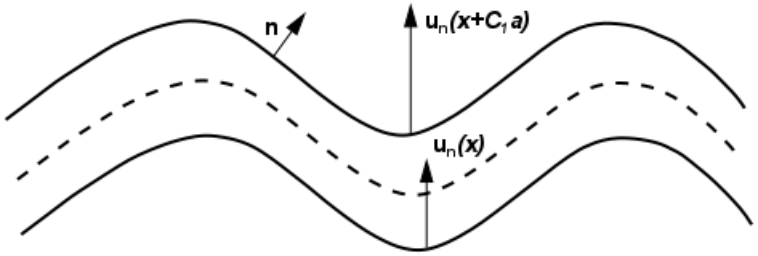
$$q_u^2 \approx \left(\frac{\pi}{24}\right)g_n L+\frac{\pi\sigma}{2L} \tag{14.476}$$

where L is a typical length scale of dominant interface features, and g_n is the scalar product of interface normal and gravity vector.

14.5.20.4. Modeling Entrainment-Absorption

Figure 14.12: Droplet Entrainment at the Interface (p. 723) shows an average location of the phase interface. x represents a point on the average interface, $\mathbf{n}$ is the inward unit normal vector, and $\mathbf{u}_n$ is the outward component of the average liquid phase velocity.

Figure 14.12: Droplet Entrainment at the Interface



The entrainment method proposed by Höhne for the AIAD approach [251] (p. 1067) assumes that the interface becomes destabilized due to turbulence, forming fluid particles with an average size of $a(x)$ in a layer close to the interface. This layer has a thickness of $C_1 a$, where C_1 is a non-dimensional parameter. The quantity of fluid passing the interfacial layer relative to the velocity of the interface affects the entrained particle formation. The entrained phase is formed from the film only if the liquid bubble moves into the gas continuous phase relative to the interface.

The roughness at the interface is calculated as:

$$a=\frac{C_2 k}{g} \tag{14.477}$$

where C_2 is a non-dimensional parameter, g is gravity, and k is the local turbulent kinetic energy, which includes the subgrid wave turbulent contribution. The AIAD approach provides a realistic modeling of the interface between two continuous fluids by taking into account the accurate modeling of subgrid turbulence phenomenon affecting the interface.

A local deposition rate can be defined as:

$$E_L(x)=\frac{C_{ent}}{g}k(x)\frac{\partial u_n}{\partial n}(x) \tag{14.478}$$

where u_n is the inward component of the average phase velocity on the interface, and $C_{ent}=C_1 C_2$ is a user-specified constant. The parameter C_{ent} depends on the fluid properties. The default value of 0.02 has shown to work for various applications that involve the AIAD model. The entrained

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

phase is distributed as a volume source at the interface in a layer with thickness Φ_{ent}. The frequency of the entrained phase formation can be expressed as:

$$E(x)=\frac{C_{ent}}{g\Phi_{ent}}k(x)\frac{\partial u_n}{\partial n}(x) \tag{14.479}$$

The turbulent kinetic energy k and the outward velocity gradient $\partial u_n/\partial n$ are the key variables to calculate the deposition rate $E(x)$. The rate of entrained liquid particles (that is, droplets) from the continuous phase is expressed as:

$$m_{c\to d}(x)=f_{FS}\alpha_c\rho_c E(x) \tag{14.480}$$

where f_{FS} is calculated by Equation 14.457 (p. 719), α_c and ρ_c are the volume fraction and the density of the secondary continuous phase, respectively.

If the entrained phase formation occurs under the continuous fluid surface, or the entrained phase comes into contact with the continuous phase of the same material, then the entrained-phase absorption results in mass transfer of the entrained droplets to the continuous liquid phase. This can be described as:

$$m_{d\to c}(x)=\frac{f_x\,\alpha_{ent}\,\rho_{ent}}{\Delta t} \tag{14.481}$$

In Equation 14.481 (p. 724), α_{ent} is the volume fraction of the entrained phase (droplets or bubbles), ρ_{ent} is the density of the entrained phase, Δt is the numerical time step defined in the simulation, and f_x is the blending function (the subscript x refers to bubbles (b) or droplets (d) depending on the entrained phase).

The entrained phase formation occurs in a layer of thickness of smeared interface $\Phi_{ent}=C_1 a=4\Delta x$ where Δx is the characteristic cell length.

Within the AIAD framework, the entrained phase complements the liquid and gas continuous phases to form a three-phase flow. The entrained phase can further grow, which can be modeled via aggregation and breakage kernels using population balance models, such as the Inhomogeneous Discrete method or Direct Quadrature Method of Moments (DQMOM) described in chapter Population Balance Model in the Ansys Fluent Theory Guide.

In Ansys Fluent, the entrainment and absorption mechanism are combined into one AIAD mass transfer mechanism. A positive mass transfer represents the entrainment of the continuous fluid into the dispersed phase, while a negative mass transfer represents the absorption of the entrained phase into the continuous fluid.

For more information on how to use the AIAD model, refer to Using the Algebraic Interfacial Area Density (AIAD) Model in the *Fluent User's Guide*.

14.5.21. Generalized Two Phase (GENTOP) Flow Model

One of the most critical weaknesses of the Euler-Euler approach for gas-liquid flows is the limited validity range of the closure laws (such as drag, lift, wall lubrication, turbulent dispersion, and turbulent interaction). The fluid particle size range of the closure relations is highly restricted by experimental data on one hand and the computational complexity of DNS simulations on the other hand. Interfacial closures for large gaseous particles, such as churn-like or Taylor bubbles, are not easy to obtain due

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

to the instability of the interface of the large bubbles rising without interference from the small bubbles. Conventional Eulerian approaches cannot be used for predictions of transitional flows in industrial applications where separated and dispersed flow structures appear simultaneously.

The GENeralized TwO Phase flow (GENTOP) method bypasses these limitations by resolving the largest fluid structures for which no closure laws exist. In essence, the GENTOP concept extends the population balance equation (PBE) method known as the inhomogeneous discrete method (iDM) by introducing a potentially continuous secondary phase. In the GENTOP method, the last velocity group defined within the iDM represents all gas structures that are larger than an equivalent spherical bubble diameter $d_{g,\max}$.

The GENTOP method is a multi-field two-fluid approach where the flow is represented by a continuous primary phase (q), one or several polydispersed secondary phases (p_i), and a GENTOP phase (pg). The term "polydispersed" refers to a wide range of size distribution. The GENTOP phase can behave as either continuous or dispersed depending on the phase volume fraction and critical bubble diameter. The dispersed phases are modeled using the iDM, which can deal with different size groups and associated velocity fields. With the iDM, various interfacial closures can be used depending on the expected size of the dispersed fields (for example, this is useful when both negative and positive lift coefficients are expected depending on the bubble size and deformability based on the Eötvös number, which determines whether the bubble moves towards the wall or the bulk of a pipe). In addition, the iDM uses appropriate models to consider transfer between different size groups due to coalescence and breakup within the polydispersed fields.

The GENTOP approach was originally introduced by the Helmholzt-Zentrum Dresden-Rossendorf (HZDR) and has been further improved, extended, and extensively validated as reported in [231] (p. 1066), [232] (p. 1066), [453] (p. 1079), [451] (p. 1079), [449] (p. 1079), and [450] (p. 1079).

The GENTOP method can model flows with bubbles smaller than the grid size and track the interface of larger continuous gaseous structures (such as churn, Taylor bubbles, and even transition to annular gas core) while accounting for mass transfer between the dispersed and the continuous gas fields.

Within the GENTOP framework, the simulation initially uses the Euler-Euler model to predict the flow behavior. Once the maximum diameter of the largest fluid structures $d_{g,\max}$ is reached, the simulation switches to the GENTOP method.

The key aspects of the GENTOP method are described in the following sections:

14.5.21.1. Interface Detection of the GENTOP Phase

14.5.21.2. Clustering Force for the GENTOP Phase

14.5.21.3. Surface Tension for the GENTOP-Primary Phase Pair

14.5.21.4. Interface Momentum Transfer

14.5.21.5. Complete Coalescence Method

14.5.21.1. Interface Detection of the GENTOP Phase

To resolve the large fluid structures, the GENTOP model detects a potential gas-liquid interface. A blending function φ_{fs} is used to identify the local interfacial structure. The free surface region is defined using the volume fraction gradient of the GENTOP phase $|\nabla \alpha_{pg}|$. The interface between the GENTOP phase and the primary phase is characterized by a variation of the volume fraction of

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$\nabla \alpha_{pg}$ from 0 to 1 over n grid cells of the size Δx. This leads to a critical value $|\nabla \alpha_{pg}|_{crit} = 1/\left(n\Delta x\right)$ which gives a definition of the interface. The free-surface detection function is then defined as:

$$\varphi_{fs} = 0.5 \tanh\left[a_{fs}\Delta x\left(|\nabla \alpha_{pg}| - |\nabla \alpha_{pg}|_{crit}\right)\right] + 0.5 \tag{14.482}$$

where a_{fs} = 100.

14.5.21.2. Clustering Force for the GENTOP Phase

The clustering force is an additional interfacial force that works exclusively between the GENTOP phase and the primary phase. It provides a smooth transition from the dispersed to continuous morphology within the GENTOP phase using an aggregative effect over the volume fraction of the continuous gas phase. While the Eulerian approach produces smearing of the volume fraction by numerical diffusion, the clustering force also stabilizes the interface without the need to use interface sharpening schemes.

The clustering force is included in the interfacial momentum transfer and is proportional to the gradient of the volume fraction of the primary phase:

$$M_{qclust} = -M_{pg}^{clust} = c_{clust}\varphi_{clust}\rho_q \nabla \alpha_q \tag{14.483}$$

where

M_q^{clust} = clustering force in the primary phase

M_{pg}^{clust} = contribution from the GENTOP phase to the clustering force

c_{clust} = clustering coefficient, which by default is equal to 1

φ_{clust} = range of applicability of the clustering force

ρ_q = primary phase density

$\nabla \alpha_q$ = gradient of the volume fraction of the primary phase

When the critical bubble size $d_{g,\max}$ specified for the GENTOP phase is reached, the clustering force begins to slowly increase creating regions of the continuous volume fraction by inducing aggregation of the GENTOP phase volume fraction until the formation of the continuous structures is completed. The clustering force is not present in the continuous structures and acts only outside of the interface agglomerating the gas when the GENTOP phase is in a dispersed state. Once the GENTOP phase reaches the critical gradient of the volume fraction, the clustering force decreases and disappears when a full interface is established.

The Ansys Fluent GENTOP model uses a modified version of the original blending function proposed in [449] (p. 1079). This modified function is based on a more physical approach to determining the regions where the clustering force should be activated.

14.5.21.3. Surface Tension for the GENTOP-Primary Phase Pair

The surface tension model creates a physical transition between the dispersed and continuous structures. This is achieved by decreasing the effect of the clustering force and including the surface tension effects as soon as the interface is detected, thus allowing deformability of the interface. The surface tension and contact angle are calculated as described in Surface Tension and Adhesion

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

for the Eulerian Multiphase Model (p. 665). In the GENTOP model, the surface tension acts only between the GENTOP and primary phase pairs.

14.5.21.4. Interface Momentum Transfer

To accurately model interfacial transfer between the dispersed and continuous morphologies of the GENTOP phase depending on the corresponding volume fraction, the GENTOP method detects the local GENTOP phase morphologies using a concept similar to that of the AIAD model ([135] (p. 1060)). (For details, see section Algebraic Interfacial Area Density (AIAD) Model in the Ansys Fluent Theory Guide.) The transition parameters between the closure models are defined in terms of the GENTOP formulation for the interfacial area density and drag and non-drag forces. The blending function for the interfacial transfers φ_{morph} is defined as [449] (p. 1079):

$$\varphi_{morph} = \begin{cases} 0 & \text{if } \varphi_{fs} < 0.5 \\ 0.5\left(\tanh\left(-5\left(0.5-\varphi_{fs}\right)\right)+0.5\right) & \text{if } 0.5 \le \varphi_{fs} < 1.0 \\ 1 & \text{if } \varphi_{fs} = 1.0 \quad \text{or} \quad \alpha_{pg} > 0.5 \end{cases} \tag{14.484}$$

Then, the interfacial transfer for the interfacial area density, drag, and non-drag forces can be defined as:

$$C_{D,pg} = \left(1-\varphi_{morph}\right) C_{D,pg_{disp}} + \varphi_{morph} C_{D,pg_{cont}} \tag{14.485}$$

$$A_{i,pg} = \left(1-\varphi_{morph}\right) A_{i,pg_{disp}} + \varphi_{morph} A_{i,pg_{cont}} \tag{14.486}$$

$$C_{ND,pg} = \left(1-\varphi_{morph}\right) C_{ND} \tag{14.487}$$

where $C_{D,pg}$, $C_{D,pg_{disp}}$, and $C_{D,pg_{cont}}$ are the drag coefficients for the GENTOP phase, dispersed part of the GENTOP phase, and continuous part of the GENTOP phase, respectively; $A_{i,pg}$, $A_{i,pg_{disp}}$, and $A_{i,pg_{cont}}$ are interfacial area densities for the GENTOP phase, dispersed part of the GENTOP phase, and continuous part of the GENTOP phase, respectively; and $C_{ND,pg}$ and C_{ND} are the non-drag forces for the GENTOP and continuous phases, respectively. The non-drag forces become zero when the GENTOP phase becomes continuous.

14.5.21.5. Complete Coalescence Method

During the calculation, low fractions of the dispersed secondary phases may arise in a region of the continuous GENTOP phase (for example, inside a churn or slug bubble). To prevent such unphysical occurrences, upon reaching the critical void fraction gradient, the GENTOP model uses a special coalescence method for complete mass transfer in the area within a fully formed interface instead of the methods for modeling aggregation due to the averaged coalescence used in the Euler-Euler approach. When certain conditions are met, the complete coalescence method converts all the remaining dispersed secondary phases within a specific grid cell into the continuous GENTOP phase. The complete coalescence method is disabled at the gas-liquid interface to allow typical coalescence and breakup at these locations.

14.5.22. The Filtered Two-Fluid Model

The kinetic theory based Two-Fluid Model (TFM) is a well-established and extensively validated model for simulating gas-particle flows in many industrial applications where it is possible to resolve all microscales present in the flow. In simulations that involve the TFM, gas-particle drag force models

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

(see Interphase Exchange Coefficients (p. 667)) along with kinetic theory based solids stress models (see Solids Pressure (p. 694) and Solids Shear Stresses (p. 697)) can predict macroscale flow features that are usually of interest.

In large-scale applications (such as commercial scale fluidized-beds), gas–particle flows are characterized by heterogeneous structures ranging from microscale to macroscale. Resolving these structures in simulations using the kinetic theory based TFM ([435] (p. 1078), [489] (p. 1081), [562] (p. 1085)) may become expensive due to the requirement for the mesh resolution to be of the order of a few particle diameters everywhere in the domain. For example, Geldart Group A particles, which fluidize well, are often used as a catalyst in many large-scale fluidized beds. The size of these particles typically ranges from 20 μm to 100 μm. Capturing microscale flow features (such as clusters and streamers in circulating fluidized bed or bubbles in bubbling fluidized bed) would require a mesh resolution of a few hundred micrometers. This would make simulations of large-scale fluidized beds prohibitively expensive.

Based on practical considerations such as computational cost, it is often desirable to employ relatively coarse meshes in a simulation. However, coarse-mesh simulations that involve the homogeneous gas-particle drag models (see Interphase Exchange Coefficients (p. 667)) do not account for the effects of microscale structures on the macroscopic flow behavior and typically over-predict fluid-particle drag force and under-predict particle phase stresses. To overcome these challenges, the filtered TFMs have been developed ([435] (p. 1078), [562] (p. 1085)). The filtered TFMs account for the effects of microscale structures and resolve macroscale structures in simulations with the focus on probing macroscale gas–particle flow features that are of principal interest in large-scale systems.

The conservation equations for filtered TFMs are well documented in literature (see, for example, [435] (p. 1078), [562] (p. 1085): The closure relations for the filtered gas-particle drag force and filtered particle phase stress used in Ansys Fluent are further described.

For the filtered drag model of Sarkar et al. [562] (p. 1085), the fluid-solid exchange coefficient is computed using Equation 14.290 (p. 679), but with index H_D calculated as:

$$H_D = 1 - \min\left[\left(a + b/\left|\tilde{v}^*_{slip}\right|\right)\bar{\alpha}_s^{\,c + d/\left|\tilde{v}^*_{slip}\right|},\ 0.97\right] \tag{14.488}$$

Here, the non-dimensional slip velocity $\tilde{v}^*_{slip}$ is defined as:

$$\tilde{v}^*_{slip} = \frac{\left|\vec{v}_s - \vec{v}_l\right|}{v_t} \tag{14.489}$$

where $\vec{v}_s$ and $\vec{v}_l$ are the velocities of solid and liquid phases, respectively, and v_t is the terminal velocity calculated by:

$$v_t = \frac{g d_s^2 (\rho_s - \rho_l)}{18 \mu_l} \tag{14.490}$$

where g is gravity, ρ_l and ρ_s are the densities of the liquid and solid phases, respectively, μ_l is the viscosity of the liquid phase, and d_s is the diameter of the solid phase particles.

Coefficients a, b, c, and d in Equation 14.488 (p. 728) have the following values:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$a=0.9506$$
$$b=0.1708$$
$$c=0.049\left(\frac{1}{\Delta^*_{filter}}-1\right)$$
$$d=0.3358$$

Here,

$$\Delta^*_{filter}=\frac{\Delta_{filter}g}{v_t^2} \tag{14.491}$$

with the filter length

$$\Delta_{filter}=2V_{cell}^{1/3} \tag{14.492}$$

where V_{cell} is the volume of the mesh cell.

The filtered solid pressure is calculated as:

$$p_s=\rho_s\Delta^2_{filter}\bar{S}_s^2\left(0.019\left(\Delta^*_{filter}\right)^2+0.006\Delta^*_{filter}+0.046\right)\bar{\alpha}_s^{\left(0.27\Delta^*_{filter}+0.9\right)} \tag{14.493}$$

where ρ_s is the density of the solid phase, $\bar{S}_s$ is the scalar filtered rates of strain in the solid phase, $\bar{\alpha}_s$ is the filtered solid volume fraction, and Δ^*_{filter} is defined by Equation 14.491 (p. 729).

The filtered granular viscosity is modeled as:

$$\bar{\mu}_s=\rho_s\left(\frac{g}{v_t^2}\right)^{-6/7}\Delta^{8/7}_{filter}\bar{S}_s\frac{0.00307\,\bar{\alpha}_s^{1.544}}{\bar{\alpha}_{s,\max}-\bar{\alpha}_s} \tag{14.494}$$

where $\bar{\alpha}_{s,\max}$ is the maximum filtered solid volume fraction.

The radial distribution is not used in the filtered TFM.

Filtered gas-phase stress models can be implemented through user-defined functions.

14.5.23. Dense Discrete Phase Model

In the standard formulation of the Lagrangian multiphase model, described in Discrete Phase (p. 447), the assumption is that the volume fraction of the discrete phase is sufficiently low: it is not taken into account when assembling the continuous phase equations. The general form of the mass and momentum conservation equations in Ansys Fluent is given in Equation 14.495 (p. 729) and Equation 14.496 (p. 729) (and also defined in Continuity and Momentum Equations (p. 2)):

$$\frac{\partial\rho}{\partial t}+\nabla\cdot\left(\rho\vec{v}\right)=S_{DPM}+S_{other} \tag{14.495}$$

$$\frac{\partial\rho\vec{v}}{\partial t}+\nabla\cdot\left(\rho\vec{v}\vec{v}\right)=-\nabla p+\nabla\cdot\tau+\rho\vec{g}+\vec{F}_{DPM}+\vec{F}_{other} \tag{14.496}$$

To overcome this limitation of the Lagrangian multiphase model, the volume fraction of the particulate phase is accounted for by extending Equation 14.495 (p. 729) and Equation 14.496 (p. 729) to the following set of equations (see also Conservation Equations (p. 662), written for phase p):

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$\frac{\partial}{\partial t}\left(\alpha_p \rho_p\right) + \nabla \cdot \left(\alpha_p \rho_p \vec{v}_p\right) = \sum_{q=1}^{nphases} \left(\dot{m}_{qp} - \dot{m}_{pq}\right) \tag{14.497}$$

$$\begin{aligned}\frac{\partial}{\partial t}\left(\alpha_p \rho_p \vec{v}_p\right) + \nabla \cdot \left(\alpha_p \rho_p \vec{v}_p \vec{v}_p\right) = &- \alpha_p \nabla p + \nabla \cdot \left[\alpha_p \mu_p \left(\nabla \vec{v}_p + \nabla \vec{v}_p^{\,T}\right)\right] \\ &+ \alpha_p \rho_p \vec{g} + F_{vm,lift,user} \\ &+ \sum_{q=1}^{nphases} \left(\vec{K}_{qp}\left(\vec{v}_q - \vec{v}_p\right) + \dot{m}_{pq}\vec{v}_{pq} - \dot{m}_{qp}\vec{v}_{qp}\right) \\ &+ K_{DPM}\left(\vec{v}_{DPM} - \vec{v}_p\right) + S_{DPM,explicit}\end{aligned} \tag{14.498}$$

Here, Equation 14.497 (p. 730) is the mass conservation equation for an individual phase p and Equation 14.498 (p. 730) is the corresponding momentum conservation equation. The momentum exchange terms (denoted by DPM) are split into an explicit part, $S_{DPM,\,explicit}$, and an implicit part, in which v_{DPM} represents the particle averaged velocity of the considered discrete phase, and K_{DPM} represents its particle averaged interphase momentum exchange coefficient [528] (p. 1083). Currently, these momentum exchange terms are considered only in the primary phase equations.

In the resulting set of equations (one continuity and one momentum conservation equation per phase), those corresponding to a discrete phase are not solved. The solution, such as volume fraction or velocity field, is taken from the Lagrangian tracking solution.

In versions prior to Ansys Fluent 13, one drawback of the dense discrete phase model was that it did not prevent the actual concentration of particles from becoming unphysically high. Hence, the model had only a limited applicability to flows close to the packing regime, such as fluidized bed reactors. To overcome this, Ansys Fluent now applies a special treatment to the particle momentum equation as soon as the particle volume fraction exceeds a certain user specified limit. Thus, the unlimited accumulation of particles is prevented. In turn, this will allow you to simulate suspensions and flows like bubbling fluidized bed reactors, operating at the packing limit conditions, allowing for polydispersed particle systems. However, no Discrete Element Method (DEM) type collision treatment is applied, which would otherwise allow the efficient simulation of systems with a large number of particles.

In the context of the phase-coupled SIMPLE algorithm (Solution Method in Ansys Fluent (p. 716)) and the coupled algorithm for pressure-velocity coupling (Selecting the Pressure-Velocity Coupling Method) in the User's Guide), a higher degree of implicitness is achieved in the treatment of the drag coupling terms. All drag related terms appear as coefficients on the left hand side of the linear equation system.

For multiphase flows with energy transfer, the enthalpy equation for the p^{th} phase has the following form:

$$\begin{aligned}\frac{\partial}{\partial t}\left(\alpha_p \rho_p \left(e_p + \frac{\vec{v}_p^{\,2}}{2}\right)\right) + \nabla \cdot \left(\alpha_p \rho_p \vec{v}_p \left(h_p + \frac{\vec{v}_p^{\,2}}{2}\right)\right) = &H_{conv} + H_{diff} + H_{visc} + H_{user} + p\frac{\partial \alpha_p}{\partial t} \\ &+ \sum_{q=1}^{nphases} \left(Q_{qp} + \dot{m}_{pq} h_{pq} - \dot{m}_{qp} h_{qp}\right) \\ &+ Q_{DPM} + S_{DPM,explicit}\end{aligned} \tag{14.499}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

The interphase heat exchange terms (denoted by DPM), like the momentum equation, are split into an explicit part, $S_{DPM,explicit}$, and an implicit part, Q_{DPM}. The latter term is assumed to be a function of the temperature difference between the phases and the interfacial area, A_i.

$$Q_{DPM}=h_{DPM}A_i\left(T_{DPM}-T_p\right) \tag{14.500}$$

T_{DPM} is the particle averaged temperature of the considered discrete phase, and h_{DPM} represents the particle-averaged interphase heat exchange coefficient. More information on all available heat exchange coefficients in DDPM context can be found in Inert Heating or Cooling (Law 1/Law 6) (p. 463). These interphase energy exchange terms are considered only in the primary phase enthalpy equation.

All other terms in Equation 14.499 (p. 730) correspond to heat transfer phenomena as described in Equation 14.197 (p. 663).

14.5.23.1. Limitations

Since the given approach makes use of the Eulerian multiphase model framework, all its limitations are adopted:

- The turbulence models: LES, SAS, DES, SDES, and SBES turbulence models are not available.
- The combustion models: PDF Transport model, Premixed, Non-premixed and partially premixed combustion models are not available.
- The solidification and melting models are not available.
- The Wet Steam model is not available.
- The real gas model (pressure-based and density-based) is not available.
- The density-based solver and models dependent on it are not available.
- DPM with the shared memory option is disabled.

14.5.23.2. Granular Temperature

The solids stress acting on particles in a dense flow situation is modeled via an additional force in the particle force balance Equation 12.1 (p. 448).

$$m_p\frac{d\vec{u}_p}{dt}=m_p\frac{\vec{u}-\vec{u}_p}{\tau_r}+m_p\frac{\vec{g}\left(\rho_p-\rho\right)}{\rho_p}+\vec{F}+\vec{F}_{interaction} \tag{14.501}$$

The term $\vec{F}_{interaction}$ models the additional force acting on a particle, resulting from interparticle interaction. It is computed from the stress tensor given by the Kinetic Theory of Granular Flows as

$$\vec{F}_{interaction}=-m_p\frac{1}{\rho_p}\nabla\cdot\overline{\overline{\tau}}_s \tag{14.502}$$

Following the same theory, the granular temperature is used in the model for the granular stress. For a more detailed description on the Kinetic Theory of Granular Flows refer to Solids Pressure (p. 694), Maximum Packing Limit in Binary Mixtures (p. 696), and Solids Shear Stresses (p. 697).

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

The granular temperature can be estimated using any of the available options described in Granular Temperature (p. 699). By default, the algebraic formulation is used. The conservation equation for the granular temperature (kinetic energy of the fluctuating particle motion) is solved with the averaged particle velocity field. Therefore, a sufficient statistical representation of the particle phase is needed to ensure the stable behavior of the granular temperature equation. For details, refer to Conservation Equations (p. 662) – Granular Temperature (p. 699).

The main advantage over the Eulerian model is that, there is no need to define classes to handle particle size distributions. This is done in a natural way in the Lagrangian formulation [528] (p. 1083).

14.5.24. Multi-Fluid VOF Model

The multi-fluid VOF model provides a framework to couple the VOF and Eulerian multiphase models. This allows the use of discretization schemes and options suited to both sharp and dispersed interface regimes while overcoming some limitations of the VOF model that arise due to the shared velocity and temperature formulation.

Modeling a Sharp Interface Regime

For cases operating in a sharp interface regime, the Multi-Fluid VOF model provides interface sharpening schemes such as Geo-Reconstruct, CICSAM, and Compressive along with the symmetric and anisotropic drag laws.

The symmetric drag law, which is used as the default in the sharp interface regime, is isotropic and tends to approach the behavior of the VOF model for high drag coefficients. The anisotropic drag law helps to overcome a limitation of the symmetric drag law by allowing higher drag in the normal direction to the interface and lower drag in the tangential direction. This facilities continuity of normal velocity across the interface while allowing different tangential velocities at the interface.

The anisotropy is characterized by the anisotropy ratio, defined as:

$$anisotropy\ \ ratio = \frac{friction\ \ factor_{normal\ to\ interface}}{friction\ \ factor_{tangential\ to\ interface}}$$

Conceptually, this ratio can be very high, however higher values sometimes lead to numerical instability. Therefore, it is recommended that you use anisotropy ratios up to approximately 1000.

Two types of drag formulations exist within the anisotropic drag law: one that is based on the symmetric drag law and the other is based on different viscosity options.

Formulation 1

This is based on the symmetric drag law, where the effective drag coefficient in the principal direction p is described as follows:

$$K_p = K_{symmetric} \lambda_p \tag{14.503}$$

where λ_p is the friction factor vector in the principal direction. $K_{symmetric}$ is the isotropic drag coefficient obtained from the symmetric drag law.

Formulation 2

The effective drag coefficient in the principal direction p is described as follows:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$K_p=\left(K_{visc}\right)_p\alpha_i\ \alpha_j \tag{14.504}$$

where α_i is the volume fraction for phase i and α_j is the volume fraction for phase j.

The viscous drag component in the principal direction $\left(K_{visc}\right)_p$ is

$$\left(K_{visc}\right)_p=\frac{\mu}{l_c^2}\lambda_p \tag{14.505}$$

where the viscosity options can be any one of the following:

- $\mu = 0.5\left(\mu_i+\mu_j\right)$
- $\mu = 2\mu_i\mu_j/\left(\mu_i+\mu_j\right)$
- $\mu = \mu_i\mu_j/\left(\mu_i\alpha_j+\mu_j\alpha_i\right)$
- $\mu = \mu_i\alpha_i+\mu_j\alpha_j$
- $\mu = \mu_i$
- $\mu = \mu_j$

and l_c is the length scale.

Note:

Both the symmetric and anisotropic drag laws used in the sharp interface regime are numerical drags. Therefore, droplet diameter in the symmetric law and length scale in the anisotropic law can be considered as numerical means to provide the drag.

Modeling a Sharp/Dispersed Interface Regime

For cases operating in the sharp/dispersed interface regime, the Multi-Fluid VOF model provides interface capturing schemes such as Compressive and Modified HRIC along with all options for all applicable drag laws.

To learn how to use the multi-fluid VOF model and the two drag formulations, refer to Including the Multi-Fluid VOF Model in the User's Guide.

14.5.25. Wall Boiling Models

The following topics will be discussed:

14.5.25.1. Overview of Wall Boiling Models
14.5.25.2. RPI Model
14.5.25.3. Non-equilibrium Subcooled Boiling
14.5.25.4. Critical Heat Flux
14.5.25.5. Interfacial Momentum Transfer
14.5.25.6. Mass Transfer
14.5.25.7. Turbulence Interactions

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

14.5.25.1. Overview of Wall Boiling Models

The term "subcooled boiling" is used to describe the physical situation where the wall temperature is high enough to cause boiling to occur at the wall even though the bulk volume averaged liquid temperature is less than the saturation value. In such cases, the energy is transferred directly from the wall to the liquid. Part of this energy will cause the temperature of the liquid to increase and part will generate vapor. Interphase heat transfer will also cause the average liquid temperature to increase, however, the saturated vapor will condense. Additionally, some of the energy may be transferred directly from the wall to the vapor. These basic mechanisms are the foundations of the so called Rensselaer Polytechnic Institute (RPI) models.

In Ansys Fluent, the wall boiling models are developed in the context of the Eulerian multiphase model. The multiphase flows are governed by the conservation equations for phase continuity (Equation 14.193 (p. 662)), momentum (Equation 14.194 (p. 662)), and energy (Equation 14.197 (p. 663)). The wall boiling phenomenon is modeled by the RPI nucleate boiling model of Kurual and Podowski [326] (p. 1072) and an extended formulation for the departed nucleate boiling regime (DNB) by Lavieville et al [343] (p. 1073).

The wall boiling models are compatible with three different wall boundaries: isothermal wall, specified heat flux, and specified heat transfer coefficient (coupled wall boundary).

Specific submodels have been considered to account for the interfacial transfers of momentum, mass, and heat, as well as turbulence models in boiling flows, as described below.

To learn how to set up the boiling model, refer to Including the Boiling Model.

14.5.25.2. RPI Model

According to the basic RPI model, the total heat flux from the wall to the liquid is partitioned into three components, namely the convective heat flux, the quenching heat flux, and the evaporative heat flux:

$$\dot{q}_W = \dot{q}_C + \dot{q}_Q + \dot{q}_E \tag{14.506}$$

The heated wall surface is subdivided into area A_b, which is covered by nucleating bubbles and a portion $(1 - A_b)$, which is covered by the fluid.

- The convective heat flux $\dot{q}_C$ is expressed as

$$\dot{q}_C = h_C (T_w - T_l)(1 - A_b) \tag{14.507}$$

where h_C is the single phase heat transfer coefficient, and T_w and T_l are the wall and liquid temperatures, respectively.

- The quenching heat flux $\dot{q}_Q$ models the cyclic averaged transient energy transfer related to liquid filling the wall vicinity after bubble detachment, and is expressed as

$$\dot{q}_Q = \frac{2k_l}{\sqrt{\pi \lambda_l T}} (T_w - T_l) A_b \tag{14.508}$$

Where k_l is the conductivity, T is the periodic time, and $\lambda_l = \frac{k_l}{\rho_l C_{pl}}$ is the diffusivity.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

- The evaporative flux $\dot{q}_E$ is given by

$$\dot{q}_E = V_d N_w \rho_v h_{fv} f \tag{14.509}$$

Where V_d is the volume of the bubble based on the bubble departure diameter, N_w is the active nucleate site density, ρ_v is the vapor density, and h_{fv} is the latent heat of evaporation, and f is the bubble departure frequency. These equations need closure for the following parameters:

- **Area of Influence**

Its definition is based on the departure diameter and the nucleate site density:

$$A_b = K \frac{N_w \pi D_w^2}{4} \tag{14.510}$$

Note that in order to avoid numerical instabilities due to unbound empirical correlations for the nucleate site density, the area of influence has to be restricted. The area of influence is limited as follows:

$$A_b = \min\left(1, K \frac{N_w \pi D_w^2}{4}\right) \tag{14.511}$$

The value of the empirical constant K is usually set to 4, however it has been found that this value is not universal and may vary between 1.8 and 5. The following relation for this constant has also been implemented based on Del Valle and Kenning's findings [136] (p. 1060):

$$K = 4.8 e^{-Ja_{sub}/80} \tag{14.512}$$

and Ja_{sub} is the subcooled Jacob number defined as:

$$Ja_{sub} = \frac{\rho_l C_{pl} \Delta T_{sub}}{\rho_v h_{fv}} \tag{14.513}$$

where $\Delta T_{sub} = T_{sat} - T_l$

- **Frequency of Bubble Departure**

Implementation of the RPI model normally uses the frequency of bubble departure as the one based on inertia controlled growth (not really applicable to subcooled boiling). [116] (p. 1059)

$$f = \frac{1}{T} = \sqrt{\frac{4g(\rho_l - \rho_v)}{3\rho_l D_w}} \tag{14.514}$$

- **Nucleate Site Density**

The nucleate site density is usually represented by a correlation based on the wall superheat. The general expression is of the form

$$N_w = C^n \left(T_w - T_{sat}\right)^n \tag{14.515}$$

Here the empirical parameters from Lemmert and Chawla [352] (p. 1073) are used, where $n = 1.805$ and $C = 210$. Other formulations are also available, such as Kocamustafaogullari and Ishii [316] (p. 1071) where

$$N_w^* = f\left(\rho^*\right) r_c^{*-4.4} \tag{14.516}$$

Here

$$N_w^* = N_w D_w^2$$
$$r_c^* = 2r_c / D_w$$
$$r_c = \frac{2\sigma T_{sat}}{\rho_v h_{fv} \Delta T_w}$$
$$\rho^* = (\rho_l - \rho_v)/\rho_v$$

Where D_w is the bubble departure diameter and the density function is defined as

$$f(\rho^*) = 2.157 \times 10^{-7} \rho^{*-3.2} (1 + 0.0049\rho^*)^{4.13} \tag{14.517}$$

- **Bubble Departure Diameter**

The default bubble departure diameter for the RPI model is based on empirical correlations [652] (p. 1090) and is calculated in meters as

$$Dw = \min\left(0.0014, 0.0006 e^{-\Delta Tsub/45.0}\right) \tag{14.518}$$

while Kocamustafaogullari and Ishii [315] (p. 1071) use

$$D_w = 0.0012(\rho^*)^{0.9} 0.0208 \emptyset \sqrt{\frac{\sigma}{g(\rho_l - \rho_v)}} \tag{14.519}$$

with $\emptyset$ being the contact angle in degrees.

The bubble departure diameter in millimeters based on the Unal relationship [661] (p. 1091) is calculated as

$$D_w = 2.4210^{-5} P^{0.709} \left(\frac{a}{b\sqrt{\varphi}}\right) \tag{14.520}$$

$$a = \frac{\Delta T_{sup}}{2\rho_g H_{lv}} \sqrt{\frac{\rho_s C_{ps} k_s}{\pi}} \tag{14.521}$$

$$b = \begin{cases} \dfrac{\Delta T_{sub}}{2\left(1 - \frac{\rho_g}{\rho_l}\right)} e^{\left(\frac{\Delta T_{sub}}{3} - 1\right)} & \Delta T_{sub} \le 3 \\ \dfrac{\Delta T_{sub}}{2\left(1 - \frac{\rho_g}{\rho_l}\right)} & \Delta T_{sub} > 3 \end{cases} \tag{14.522}$$

$$\varphi = \max\left(\left(\frac{U_b}{U_0}\right)^{0.47}, 1.0\right) \tag{14.523}$$

where P is the flow pressure, $\Delta T_{sup} = T_w - T_{sat}$ is the wall superheat, H_{lv} is latent heat, U_b is the near wall bulk velocity, and $U_b = 0.61$ m/s. The subscripts s, l, and g denote the solid material, liquid, and vapor phase, respectively.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

14.5.25.3. Non-equilibrium Subcooled Boiling

When using the basic RPI model (RPI Model (p. 734)), the temperature of the vapor is not calculated, but instead is fixed at the saturation temperature. To model boiling departing from the nucleate boiling regime (DNB), or to model it up to the critical heat flux and post dry-out condition, it is necessary to include the vapor temperature in the solution process. The wall heat partition is now modified as follows:

$$\dot{q}_W=(\dot{q}_C+\dot{q}_Q+\dot{q}_E)f(\alpha_l)+(1-f(\alpha_l))\dot{q}_V+\dot{q}_G \tag{14.524}$$

Here $\dot{q}_C$, $\dot{q}_Q$, and $\dot{q}_E$ are the liquid-phase convective heat flux, quenching heat flux, and evaporation heat flux, respectively (described in detail in RPI Model (p. 734)). The extra heat fluxes are $\dot{q}_V$ representing the convective heat flux of the vapor phase, and $\dot{q}_G$ representing heat flux to any other possible gas phases in a system. These can be expressed as

$$\dot{q}_V=h_V(T_W-T_V) \tag{14.525}$$

$$\dot{q}_G=h_G(T_W-T_G) \tag{14.526}$$

Similar to the liquid phase $\dot{q}_C$, the convective heat transfer coefficients h_V and h_G are computed from the wall function formulations.

The function $f(\alpha_l)$ depends on the local liquid volume fraction with similar limiting values as the liquid volume fraction. Lavieville et al [343] (p. 1073) proposed the following expression:

$$f(\alpha_l)=\begin{cases}1-\frac{1}{2}e^{-20(\alpha_l-\alpha_{l,crit})} & \alpha_l\geq\alpha_{l,crit}\\ \frac{1}{2}\left(\frac{\alpha_l}{\alpha_{l,crit}}\right)^{20\alpha_{l,crit}} & \alpha_l\leq\alpha_{l,crit}\end{cases} \tag{14.527}$$

Here, the critical value for the liquid fraction is $\alpha_{l,crit}=0.2$.

14.5.25.4. Critical Heat Flux

In wall boiling, the critical heat flux (CHF) condition is characterized by a sharp reduction of local heat transfer coefficients and the excursion of wall surface temperatures. It occurs when heated surfaces are no longer wetted by boiling liquid with the increase of vapor content. At critical heat flux conditions, vapor replaces the liquid and occupies the space adjacent to heated walls. The energy is therefore directly transferred from the wall to the vapor. In turn, it results in rapid reduction of the heat removal ability and sharp rise of the vapor temperature, and most importantly, the wall temperatures. In addition, wall boiling departs from the nucleating boiling regime, and the multiphase flow regime changes from a bubbly flow to a mist flow.

To model the critical heat flux conditions, the basic approach adopted in Ansys Fluent is to extend the RPI model from the nucleate boiling regime to critical heat flux and post dry-out conditions, while considering the following:

- The generalized and non-equilibrium wall heat flux partition
- The flow regime transition from bubbly to mist flows

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

14.5.25.4.1. Wall Heat Flux Partition

The wall heat partition is defined in the same way as Equation 14.524 (p. 737), with the exception of the function definition. Here, the function $f(\alpha_l)$ depends on the local liquid/vapor volume fraction with the same limiting values as the liquid volume fraction, that is, between zero and one. The Lavieville et al. model expressed by Equation 14.527 (p. 737) is used for the blending function in Equation 14.524 (p. 737). The critical values for the liquid and vapor volume fractions are $\alpha_{l,crit}=0.2$ and $\alpha_{v,crit}=1-\alpha_{l,crit}=0.8$, respectively.

There are also some other functions available to define the wall heat flux partition:

- When defining wall boiling regimes, Tentner et al. [647] (p. 1090) suggested the following expression based on the vapor volume fraction:

$$f(\alpha_v)=1-f(\alpha_l)=\begin{cases}0 & \alpha_v<\alpha_{v,1}\\ \frac{1}{2}\left(1-\cos\left(\pi\frac{\alpha_v-\alpha_{v,1}}{\alpha_{v,2}-\alpha_{v,1}}\right)\right) & \alpha_{v,1}\le\alpha_v\le\alpha_{v,2}\\ 1 & \alpha_v>\alpha_{v,2}\end{cases} \tag{14.528}$$

- Taking the thin film boiling into consideration in the wall heat flux partition, Ioilev et al. [266] (p. 1068) used a linear function to extend Equation 14.528 (p. 738) to the critical heat flux condition:

$$f(\alpha_v)=\max\left(0,\min\left(\frac{\alpha_v-\alpha_{v,1}}{\alpha_{v,2}-\alpha_{v,1}},1\right)\right) \tag{14.529}$$

In Equation 14.528 (p. 738) and Equation 14.529 (p. 738), the breakpoints have been set to $\alpha_{v,1}=0.9$ and $\alpha_{v,2}=0.95$.

In Ansys Fluent, Equation 14.528 (p. 738) is chosen as the default bending function for the wall heat flux partition.

14.5.25.4.2. Coupling Between the RPI Boiling Model and the Homogeneous or Inhomogeneous Discrete PBM

Boiling mass transfer typically occurs over a range of bubble diameters rather than a single bubble size and is also often accompanied by vigorous bubble break-up and coalescence. Therefore, it is important to account for the variation in size distribution for the accurate prediction of interfacial mass transfer and bubble volume fraction. When the RPI boiling model is coupled with the Homogeneous or Inhomogeneous discrete PBM, bubble growth and collapse due to liquid-to-gas mass transfer are modeled as nucleation and growth terms in the PBM in Ansys Fluent. The total mass transfer between phases is distributed to the various bubble size classes in proportion to their interfacial area [393] (p. 1075). The wall mass transfer in the RPI model is modeled as a nucleation term in the phase containing the smallest bin, while the bulk mass transfer is modeled as bubble growth in all gaseous phases, including the nucleating phase.

14.5.25.4.3. Flow Regime Transition

When wall boiling departs from the nucleate boiling regime and reaches the critical heat flux and post dry-out conditions, the multiphase flow regime changes from a bubbly flow to a mist flow. Consequently, the liquid phase switches from the continuous phase to the dispersed phase, while the vapor phase becomes the continuous phase from the originally dispersed phase in the

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

bubbly flow regime. With the flow regime transition, the interfacial area, momentum transfer terms (drag, lift, turbulent dispersion, interfacial area, and so on), heat transfer and turbulence quantities will change accordingly.

To mimic the change of the flow regime and compute the interfacial transfers, the so-called flow regime maps, based on cross-section averaged flow parameters, are traditionally used in sub-channel one-dimensional thermal-hydraulic codes. In CFD solvers, the concept of flow regime maps has been expanded into a local, cell-based interfacial surface topology to evaluate the flow regime transitions from local flow parameters. The ensemble of all the computational cells with their usually simple local interfacial surface topologies can provide complex global topologies to represent the different flow regimes as the traditional sub-channel flow regime maps.

As a first step, this implementation adopts a simple local interfacial surface topology to control the transition smoothly from a continuous liquid bubbly flow to a continuous vapor droplet flow configuration [647] (p. 1090), [266] (p. 1068). It assumes that inside a computational cell, the local interfacial surface topology contains multi-connected interfaces, and the flow regimes are determined by a single local flow quantity — the vapor volume fraction α_v:

- Bubbly flow topology: the vapor phase is dispersed in the continuous liquid in the form of bubbles. Typically $\alpha_v \leq 0.3$
- Mist flow topology: the liquid phase is dispersed in the continuous vapor in the form of droplets. Typically $\alpha_v \geq 0.7$
- Churn flow: this is an intermediate topology between the bubbly and mist flow topology, where $0.3 < \alpha_v < 0.7$

The interfacial surface topologies are used to compute the interfacial area and interfacial transfers of momentum and heat. Introducing φ to represent interfacial quantities (interfacial area, drag, lift, turbulent drift force and heat transfer), then they are calculated using the following general form:

$$\varphi = (1 - f(\alpha_v))\varphi_{bubbly} + f(\alpha_v)\varphi_{droplet} \tag{14.530}$$

Here $f(\alpha_v)$ is computed using equation Equation 14.528 (p. 738) or Equation 14.529 (p. 738), but with different lower and upper limits of the breakpoints. Typically, the values of 0.3 and 0.7 are used and φ_{bubbly} and $\varphi_{droplet}$ are the interfacial quantities from bubbly flow and mist flow, respectively. They are calculated using the interfacial sub-models presented in Interfacial Momentum Transfer (p. 739) and The Heat Exchange Coefficient (p. 701).

It may be noted that in the boiling models, the liquid is usually defined as the first phase, and the vapor as the second phase. Once this is defined, it remains unchanged with the flow regime transition. When φ_{bubbly} and $\varphi_{droplet}$ are calculated, however, the primary or secondary phases are switched. For φ_{bubbly}, the liquid is treated as the primary phase, while the vapor is the secondary phase. Contrary to this, for $\varphi_{droplet}$, the vapor becomes the primary phase and the liquid is the secondary phase.

14.5.25.5. Interfacial Momentum Transfer

The interfacial momentum transfer may include five parts: drag, lift, wall lubrication, turbulent drift forces, and virtual mass (described in Interphase Exchange Coefficients (p. 667)). Various models are

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

available for each of these effects, some of which are specifically formulated for boiling flows. Also, user-defined options are available.

14.5.25.5.1. Interfacial Area

The interfacial area can be calculated using either a transport equation or algebraic models as described in Interfacial Area Concentration (p. 665). For boiling flows, the algebraic formulations are typically chosen. See Interfacial Area Concentration (p. 665) for model details.

14.5.25.5.2. Bubble and Droplet Diameters

14.5.25.5.2.1. Bubble Diameter

By default, Fluent uses the following model for bubble diameter as a function of the local subcooling, $\Delta T_{sub}=T_{sat}-T_l$:

$$D_b=\begin{cases}\max\left[1.0\times10^{-5},d_{\min}\exp\left(\frac{-K(\Delta T_{sub}-\Delta T_{\max})}{d_{\min}}\right)\right] & \Delta T_{sub}>13.5\ K\\ d_{\max}-K(\Delta T_{sub}-\Delta T_{\min}) & \Delta T_{sub}\le 13.5\ K\end{cases} \tag{14.531}$$

where:

$$d_{\min}=0.00015\,m$$
$$d_{\max}=0.001\,m$$
$$\Delta T_{\min}=0\,K$$
$$\Delta T_{\max}=13.5\,K$$
$$K=\frac{d_{\max}-d_{\min}}{\Delta T_{\max}-\Delta T_{\min}}$$

As an alternative, the bubble diameter, D_b, can be given by the Unal correlation [661] (p. 1091):

$$D_b=\begin{cases}0.0015 & \Delta T_{sub}<0\\ 0.0015-0.0001\Delta T_{sub} & 0\le\Delta T_{sub}\le 13.5K\\ 0.00015 & \Delta T_{sub}>13.5K\end{cases} \tag{14.532}$$

To use the Unal correlation, Equation 14.532 (p. 740) , you can use the following scheme command:

```
(rpsetvar 'mp/boiling/bubble-diameter-model 2)
```

To return to the default formulation, Equation 14.531 (p. 740) , you can use the scheme command:

```
(rpsetvar 'mp/boiling/bubble-diameter-model 1)
```

14.5.25.5.2.2. Droplet Diameter

When the flow regime transitions to mist flow, the droplet diameter can be assumed to be constant or estimated by the Kataoka-Ishii correlation [291] (p. 1069) :

$$D_d=C_{ds}\frac{\sigma}{\rho_v j_v^2}\mathrm{Re}_l^{-1/6}\mathrm{Re}_v^{2/3}\left(\frac{\rho_v}{\rho_l}\right)^{-1/3}\left(\frac{\mu_v}{\mu_l}\right)^{2/3} \tag{14.533}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

where:

C_{ds}=0.28

j_v = vapor volumetric flux (superficial velocity)

Re_l = local liquid Reynolds number

Re_v = local vapor Reynolds number

μ_l = liquid viscosity

μ_v = vapor viscosity

14.5.25.5.3. Interfacial Drag Force

The interfacial drag force is calculated using the standard model described in Interphase Exchange Coefficients (p. 667) (and defined in the context of the interfacial area from Interfacial Area Concentration (p. 665)). As described in Fluid-Fluid Exchange Coefficient (p. 667), Ansys Fluent offers several options to calculate the drag force on dispersed phases. For boiling flows, the Ishii model (Ishii Model (p. 671)) is typically chosen, though any of the models listed in Fluid-Fluid Exchange Coefficient (p. 667) are available.

14.5.25.5.4. Interfacial Lift Force

As described in Lift Force (p. 682), Ansys Fluent offers several options to calculate the lift forces on the dispersed phases. For boiling flows, this force is important in the nucleating boiling regime. In the RPI model, the Tomiyama model (Tomiyama Lift Force Model (p. 684)) is usually chosen to account for the effects of the interfacial lift force.

14.5.25.5.5. Turbulent Dispersion Force

As described in Turbulent Dispersion Force (p. 690), Ansys Fluent offers several options to calculate the turbulent dispersion force. For boiling flows, this force is important in transporting the vapor from walls to the core fluid flow regions. In the RPI model, the Lopez de Bertodano model (Lopez de Bertodano Model (p. 691)) is usually chosen to account for the effects of the turbulent dispersion force.

14.5.25.5.6. Wall Lubrication Force

As described in Wall Lubrication Force (p. 686), Ansys Fluent offers several options to calculate the wall lubrication force on the dispersed phases. For boiling flows, this force can be important in the nucleating boiling regime. In the RPI model, the Antal et al. model (Antal et al. Model (p. 687)) is usually chosen to account for the effects of the wall lubrication force.

14.5.25.5.7. Virtual Mass Force

In the wall boiling models, the virtual mass force can be modeled using the standard correlation implemented for the Eulerian multiphase model as described in Virtual Mass Force (p. 693)

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

14.5.25.6. Mass Transfer

14.5.25.6.1. Mass Transfer From the Wall to Vapor

The evaporation mass flow is applied at the cell near the wall and it is derived from the evaporation heat flux, Equation 14.509 (p. 735):

$$\dot{m}_E = \frac{\dot{q}_E}{h_{fv} + C_{p,l}\Delta T_{sub}} \tag{14.534}$$

14.5.25.6.2. Interfacial Mass Transfer

The interfacial mass transfer depends directly on the interfacial heat transfer. Assuming that all the heat transferred to the interface is used in mass transfer (that is, evaporation or condensation), the interfacial mass transfer rate can be written as:

$$\dot{m} = \dot{m}_{lt} + \dot{m}_{vt} = \frac{\dot{q}_{lt} + \dot{q}_{vt}}{h_{fv}} \tag{14.535}$$

14.5.25.7. Turbulence Interactions

To model boiling flows, turbulence interaction models are usually included in the multiphase turbulence models to describe additional bubble stirring and dissipation. As described in Turbulence Models (p. 705) , three options are available for boiling flows: Troshko-Hassan (default), Simonin et al, and Sato.

14.6. Wet Steam Model Theory

Information is organized into the following subsections:

14.6.1. Overview of the Wet Steam Model
14.6.2. Limitations of the Wet Steam Model
14.6.3. Wet Steam Flow Equations
14.6.4. Phase Change Model
14.6.5. Built-in Thermodynamic Wet Steam Properties
14.6.6. Real Gas Property (RGP) Table Files for the Wet Steam Model
14.6.7. Computing Stagnation Conditions for the Wet Steam Model

14.6.1. Overview of the Wet Steam Model

During the rapid expansion of steam, a condensation process will take place shortly after the state path crosses the vapor-saturation line. The expansion process causes the super-heated dry steam to first subcool and then nucleate to form a two-phase mixture of saturated vapor and fine liquid droplets known as wet steam.

Modeling wet steam is very important in the analysis and design of steam turbines. The increase in steam turbine exit wetness can cause severe erosion to the turbine blades at the low-pressure stages, and a reduction in aerodynamic efficiency of the turbine stages operating in the wet steam region [454] (p. 1079).

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Ansys Fluent has adopted the Eulerian-Eulerian approach for modeling wet steam flow. The flow mixture is modeled using the compressible Navier-Stokes equations, in addition to two transport equations for the liquid-phase mass-fraction (β), and the number of liquid-droplets per unit volume (η). The phase change model, which involves the formation of liquid-droplets in a homogeneous nonequilibrium condensation process, is based on the classical non-isothermal nucleation theory.

This section describes the theoretical aspects of the wet steam model. Information about enabling the model and using your own property functions and data with the wet steam model is provided in Setting Up the Wet Steam Model in the User's Guide. Solution settings and strategies for the wet steam model can be found in Wet Steam Model in the User's Guide. Postprocessing variables are described in Model-Specific Variables in the User's Guide.

The wet steam model is compatible with General Turbo Interfaces (GTI) that are used to simulate flows in rotating and stationary components of turbomachines, in particular steam turbines. Details of different GTI types are described in Blade Row Interaction Modeling in the *Fluent User's Guide*.

The Wet Steam model is compatible with Aerodynamic Damping (blade flutter) analysis for steam turbine applications. For details on aerodynamic damping analysis, see Aerodynamic Damping (Blade Flutter Analysis) in the *Fluent User's Guide*.

14.6.2. Limitations of the Wet Steam Model

The following restrictions and limitations currently apply to the wet steam model in Ansys Fluent:

- Pressure inlet, mass-flow inlet, and pressure outlet are the only inflow and outflow boundary conditions available.
- When the wet steam model is active, the access to the **Create/Edit Materials** dialog box is restricted. Therefore, if solid properties need to be set and adjusted, then it must be done in the **Create/Edit Materials** dialog box before activating the wet steam model.

14.6.3. Wet Steam Flow Equations

The wet steam is a mixture of two-phases. The primary phase is the gaseous-phase consisting of water-vapor (denoted by the subscript *v*) while the secondary phase is the liquid-phase consisting of condensed-water droplets (denoted by the subscript *l*).

The following assumptions are made in this model:

- The velocity slip between the droplets and gaseous-phase is negligible.
- The interactions between droplets are neglected.
- The mass fraction of the condensed phase, β (also known as wetness factor), is small ($\beta < 0.2$).
- Since droplet sizes are typically very small (from approximately 0.1 microns to approximately 100 microns), it is assumed that the volume of the condensed liquid phase is negligible.

From the preceding assumptions, it follows that the mixture density (ρ) can be related to the vapor density (ρ_v) by the following equation:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$\rho=\frac{\rho_v}{(1-\beta)} \tag{14.536}$$

In addition, the temperature and the pressure of the mixture will be equivalent to the temperature and pressure of the vapor-phase.

The mixture flow is governed by the compressible Navier-Stokes equations given in vector form by Equation 24.78 (p. 992):

$$\frac{\partial W}{\partial Q}\frac{\partial}{\partial t}\int_V Q dV+\oint[F-G]\cdot dA=\int_V H dV \tag{14.537}$$

where Q=(P,u,v,w,T) are mixture quantities. The flow equations are solved using the same density-based solver algorithms employed for general compressible flows.

To model wet steam, two additional transport equations are needed [268] (p. 1068). The first transport equation governs the mass fraction of the condensed liquid phase (β):

$$\frac{\partial \rho\beta}{\partial t}+\nabla\cdot\left(\rho\vec{v}\beta\right)=\Gamma \tag{14.538}$$

where Γ is the mass generation rate due to condensation and evaporation (kg per unit volume per second). The second transport equation models the evolution of the number density of the droplets per unit volume:

$$\frac{\partial \rho\eta}{\partial t}+\nabla\cdot\left(\rho\vec{v}\eta\right)=\rho I \tag{14.539}$$

where I is the nucleation rate (number of new droplets per unit volume per second).

To determine the number of droplets per unit volume, Equation 14.536 (p. 744) and the average droplet volume V_d are combined in the following expression:

$$\eta=\frac{\beta}{(1-\beta)V_d\left(\rho_l/\rho_v\right)} \tag{14.540}$$

where ρ_l is the liquid density and the average droplet volume is defined as

$$V_d=\frac{4}{3}\pi\bar{r}_d^3 \tag{14.541}$$

where r_d is the droplet radius.

Together, Equation 14.537 (p. 744), Equation 14.538 (p. 744), and Equation 14.539 (p. 744) form a closed system of equations that, along with Equation 14.536 (p. 744), permit the calculation of the wet steam flow field.

14.6.4. Phase Change Model

The following is assumed in the phase change model:

- The condensation is homogeneous (that is, no impurities present to form nuclei).
- The droplet growth is based on average representative mean radii.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

- The droplet is assumed to be spherical.
- The droplet is surrounded by infinite vapor space.
- The heat capacity of the fine droplet is negligible compared with the latent heat released in condensation.

The mass generation rate Γ in the classical nucleation theory during the nonequilibrium condensation process is given by the sum of mass increase due to nucleation (the formation of critically sized droplets) and also due to growth/demise of these droplets [268] (p. 1068).

Therefore, Γ is written as:

$$\Gamma=\frac{4}{3}\pi\rho_l I r_*^3+4\pi\rho_l\eta\bar{r}^2\frac{\partial\bar{r}}{\partial t} \tag{14.542}$$

where $\bar{r}$ is the average radius of the droplet, and r_* is the Kelvin-Helmholtz critical droplet radius, above which the droplet will grow and below which the droplet will evaporate. An expression for r_* is given by [724] (p. 1095).

$$r_*=\frac{2\sigma}{\rho_l RT\ln S} \tag{14.543}$$

where σ is the liquid surface tension evaluated at temperature T, ρ_l is the condensed liquid density (also evaluated at temperature T), and S is the super saturation ratio defined as the ratio of vapor pressure to the equilibrium saturation pressure:

$$s=\frac{P}{P_{sat}(T)} \tag{14.544}$$

The expansion process is usually very rapid. Therefore, when the state path crosses the saturated-vapor line, the process will depart from equilibrium, and the supersaturation ratio S can take on values greater than one.

The condensation process involves two mechanisms, the transfer of mass from the vapor to the droplets and the transfer of heat from the droplets to the vapor in the form of latent heat. This energy transfer relation was presented by Hill in [245] (p. 1067) and used in [268] (p. 1068) and can be written as:

$$\frac{\partial r}{\partial t}=\frac{P}{h_{lv}\rho_l\sqrt{2\pi RT}}\frac{\gamma+1}{2\gamma}C_{pv}(T_0-T) \tag{14.545}$$

where T_0 is the droplet temperature, h_{lv} is the specific enthalpy of evaporation at pressure P, C_{pv} is the vapor isobaric specific heat capacity, γ is the ratio of specific heat capacities, and R is the specific gas constant for gaseous mixture of air and vapor.

Hill's droplet growth formula (Equation 14.545 (p. 745)) is shown to predict reasonably well the Wilson Point pressure rise for nozzle flows. However, it tends to underestimate average droplet size distribution. To improve the average droplet size, Young's droplet growth formula is introduced from [722] (p. 1095). This formula is tunable with two modeling parameters, α and β :

$$\frac{\partial r}{\partial t}=\frac{k_v\Delta T\left(1-\frac{r^*}{r}\right)}{\rho_l h_{lv}\bar{r}\left(\frac{1}{1+2\beta Kn}+3.78(1-\nu)\frac{Kn}{\mathrm{Pr}}\right)} \tag{14.546}$$

where

$$\nu=\frac{RT_s}{h_{lv}}\left(\alpha-0.5-\frac{2-q_c}{2q_c}\left(\frac{\gamma+1}{2\gamma}\right)\left(\frac{C_{pv}T_s}{h_{lv}}\right)\right)$$

and ΔT is the vapor subcooling:

$$\Delta T=T_s-T$$

with T_s being the saturation temperature at pressure P.

Other variables are as follows:

$Kn=\bar{l}/2r$ is the Knudsen number, $\bar{l}=1.5\mu_v\sqrt{RT/p}$ is the mean free path of a vapor molecule, μ_v is the vapor dynamic viscosity, k_v is the vapor thermal conductivity, $\mathrm{Pr}=\mu_v c_{pv}/k_v$ is the vapor Prandtl number, q_c is the evaporation coefficient, α is a modeling parameter with default value 9. It is the growth coefficient that relates the condensation coefficient with the evaporation coefficient. β is a modeling parameter with default 1. It is a coefficient that adjusts the distance from the droplet at which continuum processes, as opposed to free-molecular processes, occur, with typical values between 0 and 2.

Both formulas for the droplet growth are available in Fluent, with Young's formula being the default.

The classical homogeneous nucleation theory describes the formation of a liquid-phase in the form of droplets from a supersaturated phase in the absence of impurities or foreign particles. The nucleation rate described by the steady-state classical homogeneous nucleation theory [724] (p. 1095) and corrected for non-isothermal effects, is given by:

$$I=\frac{q_c}{(1+\theta)}\left(\frac{\rho_v^2}{\rho_l}\right)\sqrt{\frac{2\sigma}{M_m{}^3\pi}}\,e^{-\left(\frac{4\pi r_*{}^2\sigma}{3k_BT}\right)} \tag{14.547}$$

where q_c is evaporation coefficient, k_B is the Boltzmann constant, M_m is mass of one molecule, σ is the liquid surface tension, and ρ_l is the liquid density at temperature T.

A non-isothermal correction factor, θ, is given by:

$$\theta=\frac{2(\gamma-1)}{(\gamma+1)}\left(\frac{h_{lv}}{RT}\right)\left(\frac{h_{lv}}{RT}-0.5\right) \tag{14.548}$$

where h_{lv} is the specific enthalpy of evaporation at pressure p and γ is the ratio of specific heat capacities.

14.6.5. Built-in Thermodynamic Wet Steam Properties

There are many equations that describe the thermodynamic state and properties of steam. While some of these equations are accurate in generating property tables, they are not suitable for fast CFD computations. Therefore, Ansys Fluent uses simpler forms of the thermodynamic state equations for efficient CFD calculations that are accurate over a wide range of temperatures and pressures. Two alternative formulations for the equation of state that are used in the solver are taken from [58] (p. 1056) and [723] (p. 1095), and they are described below along with the equations of steam properties.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

14.6.5.1. Equation of State

There are two available formulations for the equation of state for steam and they are described in the following sections:

14.6.5.1.1. Virial Equation Developed by Vukalovich

14.6.5.1.2. Virial Equation Developed by Young

14.6.5.1.1. Virial Equation Developed by Vukalovich

The equations of state developed for steam in literature often refer to equilibrium states (dry steam region). To calculate properties of supercooled vapor, it is necessary to extrapolate these equations into the meta-stable region which may be beyond their range of validity. The problem was investigated in [57] (p. 1056)and it was found that the virial equation developed by Vukalovich [679] (p. 1092) could be extrapolated into these states with negligible error. It has therefore been adopted in the solver as the default option and the equation reads as:

$$p=\rho_v RT\left(1+B\rho_v+C\rho_v^2+D\rho_v^2\right) \tag{14.549}$$

In this equation, B, C and D are the second, third, and fourth virial coefficients, respectively. They are given by the following empirical functions:

$$\begin{aligned} B&=-\tfrac{e}{GT}-\phi_1+b \\ C&=-b\phi_1+4\phi_1^2\phi_2 \\ D&=32b\phi_1^2\phi_2 \end{aligned} \tag{14.550}$$

where

$$\begin{aligned} \phi_1&=\tfrac{EG}{T^{W_1}} \\ \phi_2&=1-\tfrac{K}{T^{W_2}} \end{aligned} \tag{14.551}$$

and

$$\begin{aligned} &e=63.2 \quad b=0.00085 \quad E=0.39x10^6 \quad G=47.053 \quad K=22.7 \quad n=0.355x10^{-7} \\ &m_1=1.968 \quad m_2=2.957 \quad W_1=(3+2m_1)/2 \quad W_2=(3m_2-4m_1)/2 \end{aligned} \tag{14.552}$$

The units for the variables in Equation 14.549 (p. 747) are Pa for the static pressure p, K for the static temperature T, and kg/m^3 for the vapor density ρ_v. Virial coefficients (Equation 14.550 (p. 747)) are given in the units m^3/kg, m^6/kg^2, and m^9/kg^3 for B, C, and D, respectively. The specific gas constant of water vapor R is equal to 461.52 J/kgK.

The vapor specific enthalpy h_v is given in J/kg by:

$$\begin{aligned} h_v&=\tfrac{P}{\rho_v}-RT^2\left(\tfrac{dB}{dT}\rho_v+\tfrac{1}{2}\tfrac{dC}{dT}\rho_v^2+\tfrac{1}{3}\tfrac{dD}{dT}\rho_v^3\right)+ \\ &10^3\left(1.111177T\ +\ 3.55878x10^{-4}\ T^2\ -\tfrac{6991.96}{T}\ +\ 2070.54\right) \end{aligned} \tag{14.553}$$

The vapor specific entropy s_v is given in J/kgK by:

$$\begin{aligned} s_v&=R\left(ln\tfrac{1}{\rho_v}-B\rho_v-\tfrac{1}{2}C\rho_v^2-\tfrac{1}{3}D\rho_v^3\right)-RT\left(\tfrac{dB}{dT}\rho_v+\tfrac{1}{2}\tfrac{dC}{dT}\rho_v^2+\tfrac{1}{3}\tfrac{dD}{dT}\rho_v^3\right)+ \\ &10^3\left(1.111177\ln T+7.11756x10^{-4}\,T\ -3495.98T^2\ +\ 0.30773\right) \end{aligned} \tag{14.554}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

The vapor isobaric and isochoric specific heat capacities, Cp_v and Cv_v , are given in J/kgK by:

$$\begin{aligned} Cv_v &= \frac{\partial}{\partial T}\left(h_v - \frac{P}{\rho_v}\right) \\ Cp_v &= Cv_v + \rho_v^2 T\left(\frac{\partial P}{\partial T}\right)^2 \Big/ \left(\frac{\partial P}{\partial \rho_v}\right) \end{aligned} \tag{14.555}$$

The empirical functions that define the virial coefficients and the thermodynamic properties cover the temperature range 273-973 K and the pressure range $10^3 - 10^7\ Pa$.

The vapor dynamic viscosity μ_v and thermal conductivity k_v are functions of temperature only and were obtained from [722] (p. 1095).

14.6.5.1.2. Virial Equation Developed by Young

An alternative equation of state used in the solver, which relates the pressure to the vapor density and the temperature, is given by Young [723] (p. 1095):

$$P = \rho_v RT\left(1 + B\rho_v + C{\rho_v}^2\right) \tag{14.556}$$

where B, and C are the second and the third virial coefficients given by the following empirical functions:

$$B = a_1\left(1 + \frac{\tau}{\alpha}\right)^{-1} + a_2 e^{\tau}\left(1 - e^{-\tau}\right)^{\frac{5}{2}}\tau^{-\frac{1}{2}} + a_3\tau \tag{14.557}$$

where B is given in m^3/kg, $\tau = \frac{1500}{T}$ with T given in Kelvin, α = 10000.0, a_1= 0.0015, a_2= -0.000942, and a_3= -0.0004882.

$$C = a\left(\tau - \tau_0\right)e^{-\alpha\tau} + b \tag{14.558}$$

where C is given in m^6/kg^2, $\tau = \frac{T}{647.286}$ with T given in Kelvin, τ_o= 0.8978, α=11.16, a= 1.772, and b= 1.5E-06.

The two empirical functions that define the virial coefficients B and C cover the temperature range from 273 K to 1073 K.

The vapor isobaric specific heat capacity Cp_v is given by:

$$Cp_v = Cp_0(T) + R\left[\left[\left(1 - \alpha_v T\right)\left(B - B_1\right) - B_2\right]\rho_v + \left[\left(1 - 2\alpha_v T\right)C + \alpha_v T C_1 - \frac{C_2}{2}\right]{\rho_v}^2\right] \tag{14.559}$$

The vapor specific enthalpy, h_v is given by:

$$h_v = h_0(T) + RT\left[\left(B - B_1\right)\rho_v + \left(C - \frac{C_1}{2}\right){\rho_v}^2\right] \tag{14.560}$$

The vapor specific entropy, s_v is given by:

$$s_v = s_0(T) - R\left[\ln\rho_v + \left(B + B_1\right)\rho_v + \frac{\left(C + C_1\right)}{2}{\rho_v}^2\right] \tag{14.561}$$

The isobaric specific heat at zero pressure is defined by the following empirical equation:

$$Cp_0(T) = \sum_{i=1}^{6} a_i T^{i-2} \tag{14.562}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

where Cp_0 is in KJ/kg K, a_1 = 46.0, a_2 = 1.47276, a_3 = 8.38930E-04, a_4 = -2.19989E-07, a_5 = 2.46619E-10, and a_6 = -9.70466E-14.

and

$B_1 = T\frac{dB}{dT}$, $C_1 = T\frac{dC}{dT}$, $B_2 = T^2\frac{dB^2}{dT^2}$, and $C_2 = T^2\frac{dC^2}{dT^2}$.

Both $h_0(T)$ and $s_0(T)$ are functions of temperature and they are defined by:

$$h_0(T) = \int Cp_0 dT + h_c \tag{14.563}$$

$$s_0(T) = \int \frac{Cp_0}{T} dT + s_c \tag{14.564}$$

where h_c and s_c are arbitrary constants.

The vapor dynamic viscosity μ_v and thermal conductivity k_v are also functions of temperature and were obtained from [722] (p. 1095).

14.6.5.2. Saturated Vapor Line

The saturation pressure equation as a function of temperature was obtained from [548] (p. 1085). The example provided in UDWSPF Example in the User's Guide contains a function called `wetst_satP()` that represents the formulation for the saturation pressure.

14.6.5.3. Saturated Liquid Line

At the saturated liquid-line, the liquid density, surface tension, specific heat Cp, dynamic viscosity, and thermal conductivity must be defined. The equation for liquid density, ρ_l, was obtained from [548] (p. 1085). The liquid surface tension equation was obtained from [722] (p. 1095). While the values of Cp_l, μ_l and Kt_l were curve fit using published data from [157] (p. 1062) and then written in polynomial forms. The example provided in UDWSPF Example in the User's Guide contains functions called `wetst_cpl()`, `wetst_mul()`, and `wetst_ktl()` that represent formulations for Cp_l, μ_l and Kt_l.

14.6.5.4. Mixture Properties

The mixture properties are related to vapor and liquid properties via the wetness factor using the following mixing law:

$$\varphi_m = \varphi_l \beta + (1-\beta)\varphi_v \tag{14.565}$$

where φ represents any of the following thermodynamic properties: h, s, Cp, Cv, μ or Kt.

14.6.6. Real Gas Property (RGP) Table Files for the Wet Steam Model

Alternative to the built-in thermodynamic wet steam properties are the Real Gas Property (RGP) tables that can be used to describe the thermodynamic state and properties of steam. RGP tables provide thermodynamic and transport material properties as functions of temperature and pressure. Creating

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

RGP files is describe in Using Real Gas Property (RGP) Table Files in the *Fluent User's Guide*. When creating the RGP file for the Wet Steam model, it is sufficient to generate data for only the vapor component and it will contain all the necessary information required by the model, including the superheat table, metastable vapor state, and saturation tables. The metastable data are incorporated in the superheat table section, so after crossing the vapor saturation line, the supercooled vapor properties are still available as a function of temperature and pressure. Liquid properties are taken from tables with the liquid saturation line and are functions of temperature only. Mixture properties are computed via the wetness factor using the same mixing law as given in Mixture Properties (p. 749).

14.6.7. Computing Stagnation Conditions for the Wet Steam Model

In the Wet Steam model, the values of the total pressure P_0 and total temperature T_0 are computed using the isenthalpic and isentropic relations:

$$H_0=H+\frac{1}{2}V^2,\ S_0=S \tag{14.566}$$

where the static values of the enthalpy H and entropy S are computed from the static pressure P and temperature T. If these static values of P and T are known from the solution, then the goal is to find such total values of P_0 and T_0 that $H_0(P_0,T_0)$ and $S_0(P_0,T_0)$ satisfy the equations in Equation 14.566 (p. 750).

This is done by solving the following system of equations obtained from the fundamental thermodynamic property relations:

$$\begin{aligned}\Delta H&=\left(\frac{\partial H}{\partial T}\right)_P\Delta T+\left(\frac{\partial H}{\partial P}\right)_T\Delta P\\ \Delta S&=\left(\frac{\partial S}{\partial T}\right)_P\Delta T+\left(\frac{\partial S}{\partial P}\right)_T\Delta P\end{aligned} \tag{14.567}$$

The system is solved using Newton iterations until $\Delta H=H_0(P_0,T_0)-H_0$ and $\Delta S=S_0(P_0,T_0)-S_0$ become close to zero. The resulting P_0 and T_0 are assumed to be stagnation condition values.

The same procedure is used when, on the contrary, static values of P and T have to be evaluated from P_0 and T_0, for example at a pressure inlet. In this case, the specified P_0 and T_0 are assumed to be stagnation condition values and the corresponding values of total enthalpy and entropy can be determined explicitly as $H_0(P_0,T_0)$ and $S_0(P_0,T_0)$. The isenthalpic and isentropic relations is then read as:

$$H=H_0-\frac{1}{2}V^2,\ S=S_0 \tag{14.568}$$

and the system Equation 14.567 (p. 750) is solved until $\Delta H=H(P,T)-H$ and $\Delta S=S(P,T)-S$ become close to zero. The resulting P and T are assumed to be static values.

Ansys Fluent offers two options for the treatment of enthalpy and entropy when computing stagnation conditions values. One option is to compute H and S for the mixture of vapor and liquid droplets, the other is to use the vapor (gas) phase only. The latter is assumed in Ansys CFX and is referred to as the frozen assumption for dispersed phases. In Fluent, selection of either will give different total values in the regions with non-zero wetness, as well as different static values at pressure inlets unless zero wetness is specified there.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

14.7. Modeling Mass Transfer in Multiphase Flows

This section describes the modeling of mass transfer in the framework of Ansys Fluent's general multiphase models (that is, Eulerian multiphase, mixture multiphase, VOF multiphase). There are numerous kinds of mass transfer processes that can be modeled in Ansys Fluent. You can use models available in Ansys Fluent (for example Ansys Fluent's cavitation model), or define your own mass transfer model via user-defined functions. See UDF-Prescribed Mass Transfer (p. 753) and the Fluent Customization Manual for more information about the modeling of mass transfer via user-defined functions.

Information about mass transfer is presented in the following subsections:

14.7.1. Source Terms due to Mass Transfer
14.7.2. Unidirectional Constant Rate Mass Transfer
14.7.3. UDF-Prescribed Mass Transfer
14.7.4. Cavitation Models
14.7.5. Evaporation-Condensation Model
14.7.6. Semi-Mechanistic Boiling Model
14.7.7. Interphase Species Mass Transfer

14.7.1. Source Terms due to Mass Transfer

Ansys Fluent adds contributions due to mass transfer only to the momentum, species, and energy equations. No source term is added for other scalars such as turbulence or user-defined scalars.

Let $m_{p^i q^j}$ be the mass transfer rate per unit volume from the i^{th} species of phase p to the j^{th} species of phase q. In case a particular phase does not have a mixture material associated with it, the mass transfer will be with the bulk phase.

14.7.1.1. Mass Equation

The contribution to the mass source for phase p in a cell is

$$m_p = -m_{p^i q^j} \tag{14.569}$$

and for phase q is

$$m_q = m_{p^i q^j} \tag{14.570}$$

14.7.1.2. Momentum Equation

For VOF or mixture models, there is no momentum source.

For the Eulerian model, the momentum source in a cell for phase p is

$$m_p \vec{u}_p = -m_{p^i q^j} \vec{u}_p \tag{14.571}$$

and for phase q is

$$m_q \vec{u}_q = m_{p^i q^j} \vec{u}_p \tag{14.572}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

14.7.1.3. Energy Equation

For all multiphase models, the following energy sources are added.

The energy source in a cell for phase p is

$$H_p = -\, m_{p^i q^j}\left(h_q^{f^j} - h_p^{f^i}\right) \tag{14.573}$$

and for phase q is

$$H_q = m_{p^i q^j}\left(h_q^{f^j} - h_p^{f^i}\right) \tag{14.574}$$

where $h_p^{f^i}$ and $h_q^{f^j}$ are the formation enthalpies of species i of phase p and species j of phase q, respectively. The difference between $h_p^{f^i}$ and $h_q^{f^j}$ is the latent heat.

14.7.1.4. Species Equation

The species source in a cell for species i of phase p is

$$m_p^i = -m_{p^i q^j} \tag{14.575}$$

and for species j of phase q is

$$m_q^j = m_{p^i q^j} \tag{14.576}$$

14.7.1.5. Other Scalar Equations

No source/sink terms are added for turbulence quantities and other scalars. The transfer of these scalar quantities due to mass transfer could be modeled using user-defined source terms.

14.7.2. Unidirectional Constant Rate Mass Transfer

The unidirectional mass transfer model defines a positive mass flow rate per unit volume from phase p to phase q:

$$\dot{m}_{pq} = \max\left[0, \lambda_{pq}\right] - \max\left[0, -\lambda_{pq}\right] \tag{14.577}$$

where

$$\lambda_{pq} = \dot{r}\alpha_p \rho_q \tag{14.578}$$

and $\dot{r}$ is a constant rate of particle shrinking or swelling, such as the rate of burning of a liquid droplet (units of 1/ time unit).

If phase p is a mixture material and a mass transfer mechanism is defined for species i of phase p, then

$$\lambda_{pq} = \dot{r}\alpha_p y_{p,i} \rho_q \tag{14.579}$$

where $y_{p,i}$ is the mass fraction of species i in phase p.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

14.7.3. UDF-Prescribed Mass Transfer

Because there is no universal model for mass transfer, Ansys Fluent provides a UDF that you can use to input models for different types of mass transfer, for example, evaporation, condensation, boiling, and so on. Note that when using this UDF, Ansys Fluent will automatically add the source contribution to all relevant momentum and scalar equations. This contribution is based on the assumption that the mass "created" or "destroyed" will have the same momentum and energy of the phase from which it was created or destroyed. If you would like to input your source terms directly into momentum, energy, or scalar equations, then the appropriate path is to use UDFs for user-defined sources for all equations, rather than the UDF for mass transfer. See the Fluent Customization Manual for more information about UDF-based mass transfer in multiphase.

14.7.4. Cavitation Models

A liquid at constant temperature can be subjected to a decreasing pressure, which may fall below the saturated vapor pressure. The process of rupturing the liquid by a decrease of pressure at constant temperature is called cavitation. The liquid also contains the micro-bubbles of noncondensable (dissolved or ingested) gases, or nuclei, which under decreasing pressure may grow and form cavities. In such processes, very large and steep density variations happen in the low-pressure/cavitating regions.

This section provides information about the following cavitation models used in Ansys Fluent.

- Zwart-Gerber-Belamri model [738] (p. 1095)
- Schnerr and Sauer model [582] (p. 1086) (default)

In addition, Ansys Fluent provides an expert cavitation model by Singhal et al. [597] (p. 1087), which has been available since release 6.1. However, due to convergence challenges, this model is not currently recommended. See Expert Cavitation Model (p. 758) for the description of this model.

The following assumptions are made in the standard two-phase cavitation models:

- The system under investigation must consist of a liquid and a vapor phase.
- Mass transfer between the liquid and vapor phase is assumed to take place. Both bubble formation (evaporation) and collapse (condensation) are taken into account in the cavitation models.
- For the cavitation model, Ansys Fluent defines positive mass transfer as being from the liquid to the vapor.
- The cavitation models are based on the Rayleigh-Plesset equation, describing the growth of a single vapor bubble in a liquid.
- The input material properties used in the cavitation models can be constants, functions of temperature or user-defined.

14.7.4.1. Limitations of the Cavitation Models

The following limitations apply to the cavitation models in Ansys Fluent:

- The Zwart-Gerber-Belamri and Schnerr and Sauer models do not take the effect of noncondensable gases into account by default.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

- Explicit VOF with cavitation is not recommended for the following reasons:
  - Explicit VOF is solved once per time step, therefore the mass transfer rate (which is a function of solution variables such as pressure, temperature, and volume fraction) is only evaluated once per time step. This may not give results that are consistent with those achieved using the implicit formulation. It is recommended that the volume fraction be solved every iteration, which can be enabled through the expert options in the **Multiphase Model** dialog box.
  - Explicit VOF may not be stable with sharpening schemes such as Geo-Reconstruct and CICSAM. You should use either a Sharp/Dispersed or Dispersed interface modeling option, which provide diffusive interface-capturing discretization schemes for volume fraction, such as **QUICK**, **HRIC**, and **Compressive**.
  - Explicit VOF does not model any cavitation-specific numerical treatments for added stability.
- When using the implicit VOF formulation, numerical diffusion caused by turbulent effects is added by default. This added diffusion increases the solution stability, but has adverse effects on the interfacial accuracy. If interfacial sharpness is an area of concern, you can disable the diffusion as described in Cavitation Mechanism in the *Fluent User's Guide*.

14.7.4.2. Vapor Transport Equation

With the multiphase cavitation modeling approach, a basic two-phase cavitation model consists of using the standard viscous flow equations governing the transport of mixture (Mixture model) or phases (Eulerian multiphase), and a conventional turbulence model (k- ε model). In cavitation, the liquid-vapor mass transfer (evaporation and condensation) is governed by the vapor transport equation:

$$\frac{\partial}{\partial t}\left(\alpha\rho_v\right)+\nabla.\left(\alpha\rho_v\vec{V}_v\right)=R_e-R_c \tag{14.580}$$

where

v = vapor phase

α = vapor volume fraction

ρ_v = vapor density

$\vec{V}_v$ = vapor phase velocity

R_e, R_c = mass transfer source terms connected to the growth and collapse of the vapor bubbles respectively

In Equation 14.580 (p. 754), the terms R_e and R_c account for the mass transfer between the liquid and vapor phases in cavitation. In Ansys Fluent, they are modeled based on the Rayleigh-Plesset equation describing the growth of a single vapor bubble in a liquid.

14.7.4.3. Bubble Dynamics Consideration

In most engineering situations we assume that there are plenty of nuclei for the inception of cavitation. Therefore, our primary focus is on proper accounting of bubble growth and collapse. In a flowing liquid with zero velocity slip between the fluid and bubbles, the bubble dynamics equation can be derived from the generalized Rayleigh-Plesset equation as [78] (p. 1057)

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$\Re_b \frac{D^2\Re_b}{Dt^2} + \frac{3}{2}\left(\frac{D\Re_b}{Dt}\right)^2 = \left(\frac{P_b - P}{\rho_l}\right) - \frac{4\nu_\ell}{\Re_b}\dot{\Re}_b - \frac{2\sigma}{\rho_l \Re_b} \tag{14.581}$$

where,

$\Re_b$ = bubble radius

σ = liquid surface tension coefficient

ρ_l = liquid density

ν_ℓ = liquid kinematic viscosity

P_b = bubble surface pressure

P = local far-field pressure

Neglecting the second-order terms and the surface tension force, Equation 14.581 (p. 755) is simplified to

$$\frac{D\Re_B}{Dt} = \sqrt{\frac{2}{3}\frac{P_b - P}{\rho_l}} \tag{14.582}$$

This equation provides a physical approach to introduce the effects of bubble dynamics into the cavitation model. It can also be considered to be an equation for void propagation and, hence, mixture density.

14.7.4.4. Schnerr and Sauer Model

To derive an expression of the net phase change rate, Schnerr and Sauer [582] (p. 1086) use the following two-phase continuity equations:

Liquid phase:

$$\frac{\partial}{\partial t}\left[(1-\alpha)\rho_l\right] + \nabla\cdot\left[(1-\alpha)\rho_l\vec{V}\right] = -R \tag{14.583}$$

Vapor phase:

$$\frac{\partial}{\partial t}\left(\alpha\rho_v\right) + \nabla\cdot\left(\alpha\rho_v\vec{V}\right) = R \tag{14.584}$$

Mixture:

$$\frac{\partial}{\partial t}\left(\rho\right) + \nabla\cdot\left(\rho\vec{V}\right) = 0 \tag{14.585}$$

where,

l = liquid phase

ρ = mixture density (function of phase volume fraction and density)

Mixture density ρ is defined as

$$\rho = \alpha\rho_v + (1-\alpha)\rho_l \tag{14.586}$$

Combining Equation 14.583 (p. 755), Equation 14.584 (p. 755), and Equation 14.585 (p. 755) yields a relationship between the mixture density and vapor volume fraction (α):

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$\frac{D\rho}{Dt} = -\left(\rho_l - \rho_v\right)\frac{D\alpha}{Dt} \tag{14.587}$$

The vapor volume fraction (α) can be related to the bubble number density (n) and the radius of bubble ($\mathfrak{R}_B$) as

$$\alpha = \frac{n\frac{4}{3}\pi\mathfrak{R}_B^3}{1+n\frac{4}{3}\pi\mathfrak{R}_B^3} \tag{14.588}$$

Using the Equation 14.582 (p. 755), and combining Equation 14.583 (p. 755), Equation 14.584 (p. 755), Equation 14.587 (p. 756), and Equation 14.588 (p. 756), the expression for the net phase change rate (R) is finally obtained as

$$R = \frac{\rho_v\rho_l}{\rho}\alpha\left(1-\alpha\right)\frac{3}{\mathfrak{R}_B}\sqrt{\frac{2}{3}\frac{\left(P_v - P\right)}{\rho_l}} \tag{14.589}$$

$$\mathfrak{R}_B = \left(\frac{\alpha}{1-\alpha}\frac{3}{4\pi}\frac{1}{n}\right)^{\frac{1}{3}} \tag{14.590}$$

where,

R = mass transfer rate

$\mathfrak{R}_B$ = bubble radius

Comparing Equation 14.589 (p. 756) with Equation 14.602 (p. 759) and Equation 14.595 (p. 757), it is obvious that unlike the two previous models, the mass transfer rate in the Schnerr and Sauer model is proportional to $\alpha_v\left(1-\alpha_v\right)$. Moreover, the function $f\left(\alpha_v,\rho_v,\rho_l\right) = \frac{\rho_v\rho_l}{\rho}\alpha\left(1-\alpha\right)$ has the interesting property that it approaches zero when $\alpha=0$ and $\alpha=1$, and reaches the maximum in between. Also in this model, the only parameter that must be determined is the number of spherical bubbles per volume of liquid. If you assume that no bubbles are created or destroyed, the bubble number density would be constant. The initial conditions for the nucleation site volume fraction and the equilibrium bubble radius would therefore be sufficient to specify the bubble number density (n) from Equation 14.588 (p. 756) and then the phase transition by Equation 14.589 (p. 756).

The final form of the model is as follows:

When $P_v \geq P$,

$$R_e = F_{vap}\frac{\rho_v\rho_l}{\rho}\alpha\left(1-\alpha\right)\frac{3}{\mathfrak{R}_B}\sqrt{\frac{2}{3}\frac{\left(P_v - P\right)}{\rho_l}} \tag{14.591}$$

When $P_v \leq P$,

$$R_c = F_{cond}\frac{\rho_v\rho_l}{\rho}\alpha\left(1-\alpha\right)\frac{3}{\mathfrak{R}_B}\sqrt{\frac{2}{3}\frac{\left(P - P_v\right)}{\rho_l}} \tag{14.592}$$

where F_{vap} and F_{cond} are the empirical calibration coefficients of evaporation and condensation with the default value of 1 and 0.2, respectively. You can change these coefficients using the expert text command option as described in Mass Transfer Mechanisms in the *Fluent User's Guide*.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

14.7.4.5. Zwart-Gerber-Belamri Model

Assuming that all the bubbles in a system have the same size, Zwart-Gerber-Belamri [738] (p. 1095) proposed that the total interphase mass transfer rate per unit volume (R) is calculated using the bubble number densities (n), and the mass change rate of a single bubble:

$$R=n\times\left(4\pi\mathfrak{R}_B^{\,2}\rho_v\frac{D\mathfrak{R}_B}{Dt}\right) \tag{14.593}$$

The vapor volume fraction α is related to the bubble number density and the radius of bubble as:

$$\alpha=n\times\left(\frac{4}{3}\pi\mathfrak{R}_B^{\,3}\right) \tag{14.594}$$

Substituting the value of n in Equation 14.593 (p. 757) into Equation 14.594 (p. 757), we have the expression of the net mass transfer:

$$R=\frac{3\alpha\rho_v}{\mathfrak{R}_B}\sqrt{\frac{2}{3}\frac{P_B-P}{\rho_l}} \tag{14.595}$$

Comparing Equation 14.595 (p. 757) and Equation 14.602 (p. 759), you will notice that the difference is only in the density terms in the mass transfer rate. In Equation 14.595 (p. 757), the unit volume mass transfer rate is only related to the vapor phase density (ρ_v). Unlike Equation 14.602 (p. 759), R has no relation with the liquid phase and mixture densities in this model.

As in Equation 14.602 (p. 759), Equation 14.595 (p. 757) is derived assuming bubble growth (evaporation). To apply it to the bubble collapse process (condensation), the following generalized formulation is used:

$$R_e=F\frac{3\alpha\rho_v}{\mathfrak{R}_B}\sqrt{\frac{2}{3}\frac{|P_B-P|}{\rho_l}}\,sign\left(P_B-P\right) \tag{14.596}$$

where F is an empirical calibration coefficient. Though it is originally derived from evaporation, Equation 14.596 (p. 757) only works well for condensation. It is physically incorrect and numerically unstable if applied to evaporation. The fundamental reason is that one of the key assumptions is that the cavitation bubble does not interact with each other. This is plausible only during the earliest stage of cavitation when the cavitation bubble grows from the nucleation site. As the vapor volume fraction increases, the nucleation site density must decrease accordingly. To model this process, Zwart-Gerber-Belamri proposed to replace α_v with $\alpha_{nuc}\left(1-\alpha_v\right)$ in Equation 14.596 (p. 757). Then the final form of this cavitation model is as follows:

If $P\le P_v$

$$R_e=F_{vap}\frac{3\alpha_{nuc}\left(1-\alpha_v\right)\rho_v}{\mathfrak{R}_B}\sqrt{\frac{2}{3}\frac{P_v-P}{\rho_l}} \tag{14.597}$$

If $P\ge P_v$

$$R_c=F_{cond}\frac{3\alpha_v\rho_v}{\mathfrak{R}_B}\sqrt{\frac{2}{3}\frac{P-P_v}{\rho_l}} \tag{14.598}$$

where,

$\mathfrak{R}_B$ = bubble radius = 10^{-6}m

α_{nuc} = nucleation site volume fraction = 5×10^{-4}

F_{vap} = evaporation coefficient = 50

F_{cond} = condensation coefficient = 0.01

14.7.4.6. Expert Cavitation Model

The Singhal et al. model is an expert model intended for specific applications. It can be used to account for the effect of noncondensable gases. The mass fraction of the noncondensable gases is assumed to be a known constant.

The theoretical background for this model and its limitations are provided below.

14.7.4.6.1. Limitations of the Singhal et al. Model

The following limitations exist with the Singhal et al. model:

- The Singhal et al. model can be used only for a single cavitation process. That is, only a single liquid fluid can be subjected to cavitation when mass transfer is defined through the Singhal et al. model. This liquid can be a phase or a species in a mixture phase.
- The Singhal et al. model requires the primary phase to be a liquid and the secondary phase to be a vapor. This model is only compatible with the multiphase mixture model.
- The Singhal et al. model cannot be used with the Eulerian multiphase model.
- The Singhal et al. model is not compatible with the LES turbulence model.

14.7.4.6.2. Singhal et al. Model Theory

This cavitation model is based on the "full cavitation model", developed by Singhal et al. [597] (p. 1087). It accounts for all first-order effects (that is, phase change, bubble dynamics, turbulent pressure fluctuations, and noncondensable gases). It has the capability to account for multiphase (N-phase) flows or flows with multiphase species transport, the effects of slip velocities between the liquid and gaseous phases, and the thermal effects and compressibility of both liquid and gas phases. The cavitation model can be used with the mixture multiphase model, with or without slip velocities. However, it is always preferable to solve for cavitation using the mixture model without slip velocity; slip velocities can be turned on if the problem suggests that there is significant slip between phases.

The vapor volume fraction α is related to the bubble number density and the radius of bubble as:

$$\alpha = n\times\left(\frac{4}{3}\pi\mathfrak{R}_B^{\,3}\right) \tag{14.599}$$

The relationship between the mixture density and vapor volume fraction is given by:

$$\frac{D\rho}{Dt} = -\left(\rho_l-\rho_v\right)\left(n4\pi\right)^{\frac{1}{3}}\left(3\alpha\right)^{\frac{2}{3}}\frac{D\mathfrak{R}_B}{Dt} \tag{14.600}$$

The expression for the net phase change rate (R) is calculated by:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$R=\left(n4\pi\right)^{\frac{1}{3}}\left(3\alpha\right)^{\frac{2}{3}}\frac{\rho_v\rho_l}{\rho}\left[\frac{2}{3}\left(\frac{P_B-P}{\rho_\ell}\right)\right]^{\frac{1}{2}} \tag{14.601}$$

Here R represents the vapor generation or evaporation rate, that is, the source term R_e in Equation 14.580 (p. 754). All terms, except n, are either known constants or dependent variables. In the absence of a general model for estimation of the bubble number density, the phase change rate expression is rewritten in terms of bubble radius ($\mathfrak{R}_B$), as follows:

$$R=\frac{3\alpha}{\mathfrak{R}_B}\frac{\rho_v\rho_l}{\rho}\sqrt{\frac{2}{3}\frac{\left(P_B-P\right)}{\rho_\ell}} \tag{14.602}$$

Equation 14.602 (p. 759) indicates that the unit volume mass transfer rate is not only related to the vapor density (ρ_v), but the function of the liquid density (ρ_ℓ), and the mixture density (ρ) as well. Since Equation 14.602 (p. 759) is derived directly from phase volume fraction equations, it is exact and should accurately represent the mass transfer from liquid to vapor phase in cavitation (bubble growth or evaporation). Though the calculation of mass transfer for bubble collapse or condensation is expected to be different from that of bubble growth, Equation 14.602 (p. 759) is often used as a first approximation to model the bubble collapse. In this case, the right side of the equation is modified by using the absolute value of the pressure difference and is treated as a sink term.

It may be noted that in practical cavitation models, the local far-field pressure P is usually taken to be the same as the cell center pressure. The bubble pressure P_B is equal to the saturation vapor pressure in the absence of dissolved gases, mass transport and viscous damping, that is, $P_B=P_v$.

where,

P_B = bubble pressure

P_v = saturation vapor pressure

Based on Equation 14.602 (p. 759), Singhal et al. [737] (p. 1095) proposed a model where the vapor mass fraction is the dependent variable in the transport equation. This model accommodates also a single phase formulation where the governing equations is given by:

$$\frac{\partial}{\partial t}\left(f_v\rho\right)+\nabla.\left(f_v\rho\vec{V}_v\right)=\nabla.\left(\Gamma\nabla f_v\right)+R_e-R_c \tag{14.603}$$

where,

f_v = vapor mass fraction

f_g = noncondensable gases mass fraction

Γ = diffusion coefficient

The rates of mass exchange are given by the following equations:

If $P\le P_v$

$$R_e=F_{vap}\frac{max\left(1.0,\sqrt{k}\right)\left(1-f_v-f_g\right)}{\sigma}\rho_l\rho_v\sqrt{\frac{2}{3}\frac{\left(P_v-P\right)}{\rho_l}} \tag{14.604}$$

If $P>P_v$

$$R_c = F_{cond} \frac{max\left(1.0, \sqrt{k}\right) f_v}{\sigma} \rho_l \rho_l \sqrt{\frac{2}{3} \frac{\left(P - P_v\right)}{\rho_l}} \tag{14.605}$$

The saturation pressure is corrected by an estimation of the local values of the turbulent pressure fluctuations:

$$P_v = P_{sat} + \frac{1}{2}\left(0.39 \rho k\right) \tag{14.606}$$

The constants have the values $F_{vap} = 0.02$ and $F_{cond} = 0.01$. In this model, the liquid-vapor mixture is assumed to be compressible. Also, the effects of turbulence and the noncondensable gases have been taken into account.

14.7.4.7. Turbulence Factor

The influence of turbulence on the threshold pressure can optionally be included. In all the cavitation models, the threshold pressure is calculated from, Yang et al. [127] (p. 1060):

$$P_v = P_{sat} + \frac{1}{2}\left(\text{coeff}\right) \rho_l k_l \tag{14.607}$$

where ρ_l and k_l are the liquid phase density and turbulence kinetic energy, respectively. The recommended value for coeff is `0.39` and this value is used by default.

14.7.4.8. Additional Guidelines for the Cavitation Models

In practical applications of a cavitation model, several factors greatly influence numerical stability. For instance, the high pressure difference between the inlet and exit, large ratio of liquid to vapor density, and large phase change rates between the liquid and vapor all have unfavorable effects on solution convergence. In addition, poor initial conditions very often lead to an unrealistic pressure field and unexpected cavitating zones, which, once present, are usually difficult to correct. You may consider the following factors/tips when choosing a cavitation model and addressing potential numerical problems:

- Choice of the cavitation models

  In Ansys Fluent, there are three available cavitation models. The Zwart-Gerber-Belamri and the Schnerr and Sauer models have been implemented following an entirely different numerical procedure from the Singhal et al. model. Numerically, these two models are robust and converge quickly. It is therefore highly recommended that you should use the Schnerr and Sauer or the Zwart-Gerber-Bleamri model. The Singhal et al. model, though physically similar to the other two, is numerically less stable and more difficult to use.

- Choice of the solvers

  In Ansys Fluent, both the segregated (SIMPLE, SIMPLEC, and PISO) and coupled pressure-based solvers can be used in cavitation. As usual, the coupled solver is generally more robust and converges faster, particularly for cavitating flows in rotating machinery (liquid pumps, inducers, impellers, etc). For fuel injector applications, however, the segregated solver also performs very well with the Schnerr and Sauer and the Zwart-Gerber-Belamri models.

  As for the Singhal et al. model, since the coupled solver does not show any significant advantages, it is suggested that the segregated solver is used.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

- Initial conditions

  Though no special initial condition settings are required, we suggest that the vapor fraction is always set to inlet values. The pressure is set close to the highest pressure among the inlets and outlets to avoid unexpected low pressure and cavitating spots. In general, the Schnerr and Sauer and Zwart-Gerber-Belamri models are robust enough so that there is no need for specific initial conditions. But in some very complicated cases, it may be beneficial to obtain a realistic pressure field before substantial cavities are formed. This can be achieved by obtaining a converged/near-converged solution for a single phase liquid flow, and then enabling the cavitation model. Again, the Singhal et al. model is much more sensitive to initial conditions. The above mentioned treatments are generally required.

- Pressure discretization schemes

  As for general multiphase flows, it is more desirable to use the following pressure discretization schemes in cavitation applications in this order:

  – PRESTO!

  – body force weighted

  – second order

  The standard and linear schemes generally are not very effective in complex cavitating flows and you should avoid using them.

- Relaxation factors

  – Schnerr and Sauer and Zwart-Gerber-Belamri models

    The default settings generally work well. To achieve numerical efficiency, however, the following values may be recommended:

    → The relaxation factor for vapor is 0.5 or higher unless the solution diverges or all the residuals oscillate excessively.

    → The density and the vaporization mass can be relaxed, but in general set them to 1.

    → For the segregated solver, the relaxation factor for pressure should be no less than the value for the momentum equations.

    → For the coupled solver, the default value for the Courant number (200) may need to be reduced to 20-50 in some complex 3D cases.

  – Singhal et al. model

    In general, small relaxation factors are recommended for momentum equations, usually between 0.05 – 0.4. The relaxation factor for the pressure-correction equation should generally be larger than those for momentum equations, say in the range of 0.2 – 0.7. The density and the vaporization mass (source term in the vapor equation) can also be relaxed to improve convergence. Typically, the relaxation factor for density is set between the values of 0.3 and 1.0, while for the vaporization mass, values between 0.1 and 1.0 may be appropriate. For some extreme cases, even smaller relaxation factors may be required for all the equations.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

- Special tips for the Singhal et al. model

  – Noncondensable gases

    Noncondensable gases are usually present in liquids. Even a small amount (for example, 15 ppm) of noncondensable gases can have significant effects on both the physical results and the convergence characteristics of the solution. A value of zero for the mass fraction of non-condensable gases should generally be avoided. In some cases, if the liquid is purified of noncondensable gases, a much smaller value (for example, 10^{-8}) may be used to replace the default value of 1.5×10^{-5}. In fact, higher mass fractions of the noncondensable gases may in many cases enhance numerical stability and lead to more realistic results. In particular, when the saturation pressure of a liquid at a certain temperature is zero or very small, noncondensable gases will play a crucial role both numerically and physically.

  – Limits for dependent variables

    In many cases, setting the pressure upper limit to a reasonable value can help convergence greatly at the early stage of the solution. It is advisable to always limit the maximum pressure when possible. Ansys Fluent's default value for the maximum vapor pressure limit is five times the local vapor pressure with consideration of the turbulence and thermal effects. You can change the vapor pressure ratio (which is the ratio of the maximum vapor pressure limit to the local pressure) using the expert text command option as described in Mass Transfer Mechanisms in the *Fluent User's Guide*.

  – Relaxation factor for the pressure correction equation

    For cavitating flows, a special relaxation factor is introduced for the pressure correction equation. By default, this factor is set to 0.7, which should work well for most of the cases. For some very complicated cases, however, you may experience the divergence of the AMG solver. Under those circumstances, this value may be reduced to no less than 0.4. You can set the value of this relaxation factor by typing a text command. For more information, contact your Ansys Fluent support engineer.

14.7.4.9. Extended Cavitation Model Capabilities

When cavitation occurs, in many practical applications other gaseous species exist in the systems. For instance, in a ventilated supercavitating vehicle, air is injected into a liquid to stabilize or increase the cavitation along the vehicle surfaces. In some cases, the incoming flow is a mixture of a liquid and some gaseous species. To predict those types of cavitating flows, the basic two-phase cavitation model must be extended to a multiphase (N-phase) flow, or a multiphase species transport cavitation model.

14.7.4.9.1. N-phase Cavitation Models

The multiphase cavitation models are the extensions of the three basic two-phase cavitation models to multiphase flows. When modeling a N-phase flow, multiple liquid and gas phases can be added in the computational system under the following assumptions:

- Mass transfer (cavitation) only occurs between the liquid and vapor phase.
- The basic two-phase cavitation models are still used to model the phase changes between the liquid and vapor.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

- In the Singhal et al. model, the predescribed noncondensable gases can be included in the system. To exclude noncondensable gases from the system, the mass fraction must be set to 0, and the noncondensable gas must be modeled by a separate compressible gas phase.
- For non-cavitating phase (i), the general transport equation governing the vapor phase is the volume fraction equation in the Zwart-Gerber-Belamri and Schnerr and Sauer models, while in the Singhal et al. model, a mass transfer equation is solved and the vapor must be the second phase.

14.7.4.9.2. Multiphase Species Transport Cavitation Model

The detailed description of the multiphase species transport approach can be found in Modeling Species Transport in Multiphase Flows (p. 778). The multiphase species transport cavitation model can be summarized as follows:

- All the assumptions/limitations for the multiphase cavitation model apply here.
- The mass transfer between a liquid and a vapor phase/species is modeled by the basic cavitation models.
- For the phases with multiple species, the phase shares the same pressure as the other phases, but each species has its own pressure (that is, partial pressure). As a result, the vapor density and the pressure used in Equation 14.605 (p. 760) are the partial density and pressure of the vapor.

14.7.5. Evaporation-Condensation Model

Ansys Fluent uses one of two models for interphase mass transfer through evaporation-condensation. With the VOF and mixture formulations the Lee model is used. With the Eulerian formulation, you can choose between the Lee model and the Thermal Phase Change model.

14.7.5.1. Lee Model

The Lee model [348] (p. 1073) is a mechanistic model with a physical basis. It is used with the mixture and VOF multiphase models and can be selected with the Eulerian multiphase model if one of the overall interfacial heat transfer coefficient models will be used (as opposed to the two-resistance model). In the Lee model, the liquid-vapor mass transfer (evaporation and condensation) is governed by the vapor transport equation:

$$\frac{\partial}{\partial t}(\alpha_v \rho_v) + \nabla \cdot (\alpha_v \rho_v \vec{V}_v) = \dot{m}_{lv} - \dot{m}_{vl} \tag{14.608}$$

where

v = vapor phase

α_v = vapor volume fraction

ρ_v = vapor density

$\vec{V}_v$ = vapor phase velocity

$\dot{m}_{lv}$, $\dot{m}_{vl}$ = the rates of mass transfer due to evaporation and condensation, respectively. These rates use units of kg/s/m^3

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

As shown in the right side of Equation 14.608 (p. 763), Ansys Fluent defines positive mass transfer as being from the liquid to the vapor for evaporation-condensation problems.

Based on the following temperature regimes, the mass transfer can be described as follows:

If $T_l > T_{sat}$ (evaporation):

$$\dot{m}_{lv} = coeff * \alpha_l \rho_l \frac{(T_l - T_{sat})}{T_{sat}} \tag{14.609}$$

If $T_v < T_{sat}$ (condensation):

$$\dot{m}_{vl} = coeff * \alpha_v \rho_v \frac{(T_{sat} - T_v)}{T_{sat}} \tag{14.610}$$

$coeff$ is a coefficient that must be fine tuned and can be interpreted as a relaxation time. α and ρ are the phase volume fraction and density, respectively. The source term for the energy equation can be obtained by multiplying the rate of mass transfer by the latent heat.

Consider the Hertz Knudsen formula [238] (p. 1066) [313] (p. 1071), which gives the evaporation-condensation flux based on the kinetic theory for a flat interface:

$$F = \beta \sqrt{\frac{M}{2\pi R T_{sat}}} \left(P^* - P_{sat}\right) \tag{14.611}$$

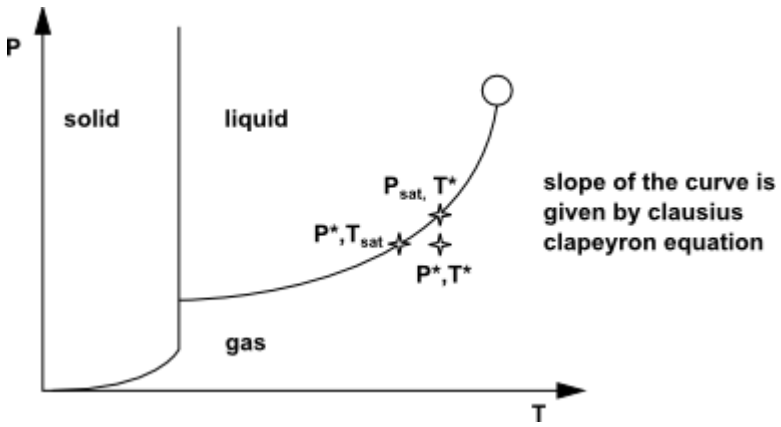
The flux has units of kg/s/m^2, P is the pressure, T is the temperature, and R is the universal gas constant. The coefficient β is the so-called accommodation coefficient that shows the portion of vapor molecules going into the liquid surface and adsorbed by this surface. P^* represents the vapor partial pressure at the interface on the vapor side. The Clapeyron-Clausius equation relates the pressure to the temperature for the saturation condition. (It is obtained by equating the vapor and liquid chemical potentials):

$$\frac{dP}{dT} = \frac{L}{T\left(v_v - v_l\right)} \tag{14.612}$$

v_v and v_l are the inverse of the density for the vapor and liquid (volume per mass unit), respectively. L is the latent heat (J/kg).

Based on this differential expression, we can obtain variation of temperature from variation of pressure close to the saturation condition.

Figure 14.13: The Stability Phase Diagram



Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

The Clausius Clapeyron equation yields the following formula as long as P^* and T^* are close to the saturation condition:

$$\left(P^*-P_{sat}\right)=-\frac{L}{T\left(v_v-v_l\right)}\left(T^*-T_{sat}\right) \tag{14.613}$$

Using this relation in the above Hertz Knudsen equation yields [642] (p. 1090)

$$F=\beta\sqrt{\frac{M}{2\pi RT_{sat}}}L\left(\frac{\rho_v\rho_l}{\rho_l-\rho_v}\right)\frac{\left(T^*-T_{sat}\right)}{T_{sat}} \tag{14.614}$$

The factor β is defined by means of the accommodation coefficient and the physical characteristics of the vapor. β approaches 1.0 at near equilibrium conditions.

In the Eulerian and mixture multiphase models, the flow regime is assumed to be dispersed. If we assume that all vapor bubbles, for example, have the same diameter, then the interfacial area density is given by the following formula:

$$A_i=\frac{6\alpha_v\alpha_l}{d_b} \tag{14.615}$$

where d_b is the bubble diameter (m) and the phase source term (kg/s/m^3) should be of the form:

$$FA_i=\frac{6}{d_b}\beta\sqrt{\frac{M}{2\pi RT_{sat}}}L\left(\frac{\alpha_v\rho_v}{\rho_l-\rho_v}\right)\left[\alpha_l\rho_l\frac{\left(T^*-T_{sat}\right)}{T_{sat}}\right] \tag{14.616}$$

From the above equation, $coeff$, which is the inverse of the relaxation time (1/s) is defined as

$$coeff=\frac{6}{d_b}\beta\sqrt{\frac{M}{2\pi RT_{sat}}}L\left(\frac{\alpha_v\rho_v}{\rho_l-\rho_v}\right) \tag{14.617}$$

This leads to the final expression for the evaporation process defined in Equation 14.609 (p. 764). It can be treated implicitly as a source term in the phase conservation equation.

A similar expression can be obtained for condensation. In this case, we consider small droplets in a continuous vapor phase even if your primary phase is a liquid.

Note that the coefficient $coeff$ should theoretically be different for the condensation and evaporation expression. Furthermore, the theoretical expression is based on a few strong assumptions:

- flat interface
- dispersed regime with constant diameter
- known β

The bubble diameter and accommodation coefficient are usually not very well known, which is why the coefficient $coeff$ must be fine tuned to match experimental data. By default, the coefficient for both evaporation and condensation is 0.1. However, in practical cases, the order of magnitude of $coeff$ can be as high 10^3.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

14.7.5.2. Thermal Phase Change Model

The thermal phase change model is available for evaporation-condensation when the Eulerian multiphase model is used with the two-resistance approach for computing phase heat transfer coefficients (Two-Resistance Model (p. 703)).

Applying Equation 14.389 (p. 703) and Equation 14.390 (p. 703) to a liquid-vapor pair gives:

- From the interface to the liquid phase:

$$Q_l = C_l h_l A_i \left(T_s - T_l \right) - \dot{m}_{lv} H_{ls} \tag{14.618}$$

- From the interface to the vapor phase:

$$Q_v = C_v h_v A_i \left(T_s - T_v \right) + \dot{m}_{lv} H_{vs} \tag{14.619}$$

where h_l and h_v are the liquid and vapor phase heat transfer coefficients, H_{ls} and H_{vs} are the liquid and vapor phase enthalpies, and C_l and C_v are the scaling factors. By default, C_l and C_v are set to 1. The interfacial temperature T_s is determined from consideration of thermodynamic equilibrium. Ignoring effects of surface tension on pressure, we can assume that $T_s = T_{sat}$, the saturation temperature.

Because neither heat nor mass can be stored on the phase interface, the overall heat balance must be satisfied:

$$Q_l + Q_v = 0 \tag{14.620}$$

From the preceding equations, the mass transfer through evaporation from the liquid to the vapor phase can be expressed as:

$$\dot{m}_{lv} = -\frac{C_l h_l A_i \left(T_{sat} - T_l \right) + C_v h_v A_i \left(T_{sat} - T_v \right)}{H_{vs} - H_{ls}} \tag{14.621}$$

The phase enthalpies, H_{vs} and H_{ls}, need to be computed correctly to take account of the discontinuity in static enthalpy due to latent heat between the two phases as well as the heat transfer from either phase to the phase interface. Using the Prakash formulation [529] (p. 1084) with the bulk fluid enthalpy carried out of the outgoing phase and the saturation enthalpy carried into the incoming phase we have:

If $\dot{m}_{lv} \geq 0$ (evaporation, the liquid phase is the outgoing phase):

$$\begin{aligned} H_{ls} &= H_l \left(T_l \right) \\ H_{vs} &= H_v \left(T_{sat} \right) \end{aligned} \tag{14.622}$$

If $\dot{m}_{lv} < 0$ (condensation, the liquid phase is the incoming phase):

$$\begin{aligned} H_{ls} &= H_l \left(T_{sat} \right) \\ H_{vs} &= H_v \left(T_v \right) \end{aligned} \tag{14.623}$$

This leads to a formulation that is stable both physically and numerically. It implies that the denominator in Equation 14.621 (p. 766) is non-zero, being greater than or equal to the latent heat:

$$L = H_v \left(T_{sat} \right) - H_l \left(T_{sat} \right) \tag{14.624}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

It may be noted that with the presence of mass transfer H_l and H_v are total phase enthalpies, which can be expressed in a general form:

$$H(T)=H(T_{ref})+\int_{T_{ref}}^{T} C_p(T)dT \tag{14.625}$$

where T_{ref} is the reference temperature and $H(T_{ref})$ is the standard state enthalpy at T_{ref}. Equation 14.625 (p. 767) applies for phases as well as species on each phase.

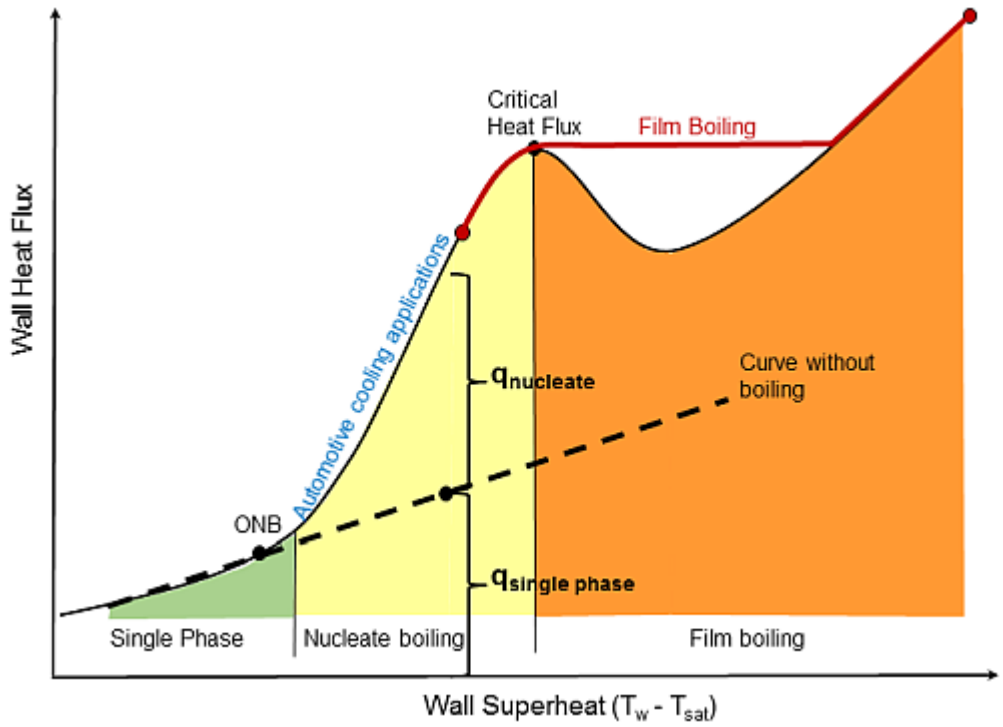
Note that in the thermal phase change model the evaporation-condensation mass transfer process is governed entirely by the interphase heat transfer processes and the overall heat balance. There is no calibration required for the mass transfer coefficients as there is in the Lee model. Therefore, it is generally recommended that you use the Eulerian multiphase formulation with the two-resistance heat transfer method when simulating evaporation-condensation processes.

14.7.6. Semi-Mechanistic Boiling Model

The study of heat transfer in boiling is important in many industrial sectors such as nuclear and automotive industries. Several boiling models have been proposed for heat transfer predictions in high-pressure applications in the nuclear industry. However, for low pressure automotive systems, high-pressure boiling models, which are based on specific experimental correlations, may be unreliable and numerically unstable.

To model sub-cooled nucleate boiling at low pressure, bulk evaporation/condensation phenomena is generally combined with the heat transfer augmentation at the wall due to boiling. In Ansys Fluent, the Lee model is used to capture the phase change process both near the wall and in the bulk domain. The heat transfer augmentation at wall is modeled by empirical correlations by Chen [133] (p. 1060).

Figure 14.14: The Boiling Curve



Chen [104] (p. 1059) proposed the flow boiling correlation for vertical tubes where the effective wall heat flux is expressed as the weighted sum of the nucleate boiling heat flux and the force convective heat flux. Chen's correlation is based on the following assumptions:

- Vapor formed by the evaporation process increases the liquid velocity, and, therefore, the convective heat transfer contribution is augmented relative to that of a single-phase liquid flow.
- The convection partially suppresses the nucleation of boiling sites and, therefore, reduces the contribution of nucleate boiling.

The effective wall heat flux q_w as proposed by Chen is expressed as:

$$q_w = F q_{sp} M_{sp} + S q_{nb} M_{nb} \tag{14.626}$$

where

F = forced convective augmentation factor

M_{sp} and M_{nb}= heat flux multipliers for the single phase and nucleate boiling, respectively

q_{sp}= single phase heat flux

q_{nb} = nucleate boiling heat flux

S = nucleate boiling suppression factor

Kutateladze [327] (p. 1072) proposed a modified form of effective heat flux q_w using the asymptotic power law, which is expressed as:

$$q_w = \left((F q_{sp} M_{sp})^n + (S q_{nb} M_{nb})^n \right)^{1/n} \tag{14.627}$$

where n is the power law superposition constant. Setting n to 1 results in a linear superposition of the single phase and the nucleate boiling heat flux, as proposed by Chen. Kutateladze proposed an asymptotic method with $n = 2$. With the increase in the value of n, the effective heat flux tends to become larger than the single phase heat flux and nucleate boiling heat flux.

The single phase heat flux q_{sp} in Equation 14.626 (p. 768) is given by:

$$q_{sp} = h_{sp} \Delta T \tag{14.628}$$

where

$$\Delta T = T_w - T_c$$

The nucleate boiling heat flux q_{nb} in Equation 14.626 (p. 768) is given by:

$$q_{nb} = h_{nb} \Delta T_{superheat} \tag{14.629}$$

where

$$T_{superheat} = T_w - T_{sat}$$

In the above equations,

h_{sp} and h_{nb}= heat transfer coefficients for the single phase and nucleate boiling, respectively

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

T_w = wall temperature

T_c = boundary cell temperature

T_{sat} = saturation temperature

The effective single phase heat transfer coefficient h_{sp} is calculated as:

$$h_{sp}=fh_l+(1-f)h_v \tag{14.630}$$

where h_l and h_v are single phase heat transfer coefficients for liquid and vapor, respectively, and f is the wetting fraction (fraction of wall wetted by liquid).

The nucleate boiling heat transfer coefficient h_{nb} is calculated using the Foster and Zuber correlation:

$$h_{nb}=0.00122\ \frac{k_l^{0.79}c_{p_l}^{0.45}\rho_l^{0.49}}{\sigma^{0.5}\mu_l^{0.29}L^{0.24}\rho_v^{0.24}}\left(P_{sat,T_w}-P_{sat,T_{sat}}\right)^{0.75}\left(T_w-T_{sat}\right)^{0.24} \tag{14.631}$$

where

k_l = thermal conductivity of liquid

c_{pl} = specific heat of liquid

ρ_l = density of liquid

μ_l = viscosity of liquid

ρ_v = density of vapor

σ = surface tension coefficient

L = latent heat

P_{sat,T_w} and $P_{sat,T_{sat}}$ = saturation pressures corresponding to wall temperature and saturation temperature, respectively

The forced convective augmentation factor F is proposed by Chen in the form of [317] (p. 1071):

$$F=2.35\left(\frac{1}{X_{tt}}+0.213\right)^{0.736}\quad for\ X_{tt}<10 \tag{14.632}$$

where the Martinelli parameter X_{tt} is used to account for the two-phase effect on convection and is defined as follows:

$$X_{tt}=\left(\frac{1-x}{x}\right)^{0.9}\left(\frac{\rho_g}{\rho_l}\right)^{0.5}\left(\frac{\mu_l}{\mu_g}\right)^{0.1} \tag{14.633}$$

$$F=1\quad for\ X_{tt}\geq 10 \tag{14.634}$$

where x is the vapor quality (mass fraction).

The nucleate boiling suppression factor S is given by:

$$S=S_{fc}S_{sub} \tag{14.635}$$

where S_{fc} is the suppression factor due to forced convection, and S_{sub} is the suppression factor due to subcooled effects. According to a proposal by Steiner et al. [630] (p. 1089), S_{sub} is calculated as:

$$S_{sub}=\frac{T_w-T_{sat}}{T_w-T_{ref}} \tag{14.636}$$

S_{fc} in Equation 14.635 (p. 769) is expressed using the formulation proposed by Butterworth et al. [89] (p. 1058):

$$S_{fc}=\begin{cases}1/\left(1+0.12Re_{TP}^{1.14}\right) & for\ Re_{TP}<32.5\\ 1/\left(1+0.42Re_{TP}^{0.78}\right) & for\ 32.5\le Re_{TP}<70\\ 0.1 & for\ Re_{TP}>70\end{cases} \tag{14.637}$$

where Re_{TP} is the modified two-phase Reynolds number scale:

$$Re_{TP}=\ 10^{-4}Re_{l,ref}F^{1.25} \tag{14.638}$$

Here, the reference Reynolds number for liquid $Re_{l,ref}$ is calculated as:

$$Re_{l,ref}=\frac{\rho_{l,T_{ref}}U_{ref}L}{\mu_{l,T_{ref}}} \tag{14.639}$$

where

$\rho_{l,T_{ref}}$ = liquid density at reference temperature T_{ref}

$\mu_{l,T_{ref}}$ = liquid viscosity at reference temperature T_{ref}

U_{ref} = reference velocity

L = length scale

For information on how to use the semi-mechanistic boiling model, see Including Semi-Mechanistic Boiling in the *Fluent User's Guide*.

14.7.7. Interphase Species Mass Transfer

You can use Fluent to model interphase species mass transfer. Interphase species mass transfer can occur across a phase interface (between a gas and a liquid, or between a liquid and a solid) depending directly on the concentration gradient of the transporting species in the phases. For example,

- evaporation of a liquid into a gaseous mixture including its vapor, such as the evaporation of liquid water into a mixture of air and water vapor.
- absorption/dissolution of a dissolved gas in a liquid from a gaseous mixture. For example, the absorption of oxygen by water from air.

The interphase species mass transfer can be solved in either the Mixture model or the Eulerian model subject to the following conditions:

- Both phases consist of mixtures with at least two species, and at least one of the species is present in both phases.
- The two mixture phases are in contact and separated by an interface.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

- Species mass transfer can only occur between the same species from one phase to the other. For example, evaporation/condensation between water liquid and water vapor.
- As in all interphase mass transfer models in Fluent, if a gas mixture is involved in an interphase species mass transfer process, it is always treated as a "to"-phase and the liquid mixture as a "from"-phase.
- For the species involved in the mass transfer, the mass fractions in both phases must be determined by solving transport equations. For example, the mass fractions of water liquid and water vapor in an evaporation/condensation case must be solved directly from the governing equations, rather than algebraically or from the physical constraint relations.

To model interphase species mass transfer, phase species transport equations are solved along with the phase mass, momentum and energy equations. The transport equation for Y_q^i, the local mass fraction of species i in phase q, is:

$$\frac{\partial\left(\alpha_q\rho_q Y_q^i\right)}{\partial t}+\nabla\cdot\left(\alpha_q\rho_q\vec{V}_q Y_q^i\right)=-\nabla\cdot\left(\alpha_q\vec{J}_q^i\right)+\alpha_q R_q^i+\ \alpha_q S_q^i+\sum_{p=1}^{n}\dot{m}_{p^iq^j}+R \tag{14.640}$$

where p denotes the pth phase, and n is the number of phases in the system. α_q, ρ_q, and $\vec{V}_q$ are the phase volume fraction, density, and velocity for the qth phase. R_q^i is the net rate of production of homogeneous species i through chemical reaction in phase q. S_q^i is the rate of production from external sources, and $\boldsymbol{R}$ is the heterogeneous reaction rate. $\dot{m}_{p^jq^i}$ denotes the mass transfer source from species j on phase p to species i on phase q.

In practice, the source term for mass transfer of a species between phase p and phase q is treated as a single term, $\dot{m}_{p^iq^j}$, depending on the direction of the mass transfer. The volumetric rate of species mass transfer is assumed to be a function of mass concentration gradient of the transported species.

$$\dot{m}_{p^iq^j}=k_{pq}A_i\left(K_{q^ip^i}^{\rho}\rho_p^j-\rho_q^i\right) \tag{14.641}$$

where k_{pq} is the volumetric mass transfer coefficient between the pth phase and the qth phase, A_i is the interfacial area, and ρ_p^j is the mass concentration of species j in phase p, and ρ_q^i is the mass concentration of species i in phase p. The derivation of the formulation of the volumetric rate of species mass transfer is presented in Modeling Approach (p. 771).

In order to solve the species mass transfer it is necessary to determine appropriate values for k_{pq} and $K_{q^ip^j}^{\rho}$.

14.7.7.1. Modeling Approach

Interphase species mass transfer is modeled using the two-resistance model in Ansys Fluent. In situations where there are discontinuities in phase concentrations at dynamic equilibrium, it is in general not possible to simulate multi-component species mass transfer with the use of a single overall mass transfer coefficient. The two-resistance model, first proposed by Whitman [704] (p. 1094), is a general approach analogous to the two-resistance heat transfer models. It considers separate mass transfer processes with different mass transfer coefficients on either side of the phase interface.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Consider a single species dissolved in the qth and pth phases with concentration of c_q^i and c_p^j, respectively, and with a phase interface, s. Interphase mass transfer from the qth phase to the pth phase involves three steps:

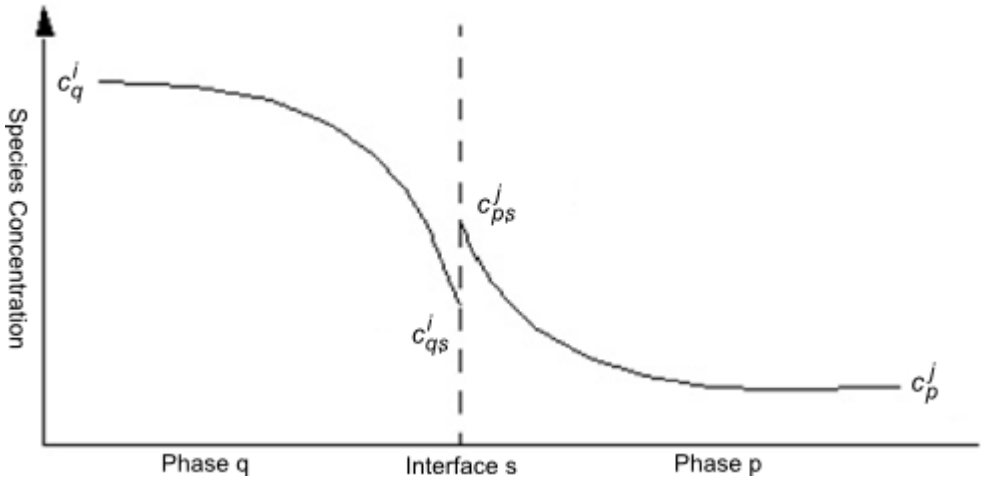
- The transfer of species i from the bulk qth phase to the phase interface, s.
- The transfer of species i across the phase interface. It is identified as species j in the pth phase.
- The transfer of species j from the interface, s, to the bulk pth phase.

The two-resistance model has two principal assumptions:

- The rate of species transport between the phases is controlled by the rates of diffusion through the phases on each side of the interface.
- The species transport across the interface is instantaneous (zero-resistance) and therefore equilibrium conditions prevail at the phase interface at all times.

In other words, there exist two "resistances" for species transport between two phases, and they are the diffusions of the species from the two bulk phases q and p to the phase interface, s. This situation is illustrated graphically in Figure 14.15: Distribution of Molar Concentration in the Two-Resistance Model (p. 772).

Figure 14.15: Distribution of Molar Concentration in the Two-Resistance Model



Using Equation 14.641 (p. 771) with the assumption that k_q and k_p are the mass transfer coefficients for the qth and pth phases, respectively, the volumetric rates of phase mass exchange can be expressed as follows:

From the interface to the qth phase,

$$\dot{m}_{qi} = k_q A_i \left(\rho_{q,s}^i - \rho_q^i \right) = k_q A_i M_i \left(c_{q,s}^i - c_q^i \right) \tag{14.642}$$

From the interface to the pth phase,

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$\dot{m}_{p^j} = k_p A_i \left(\rho^j_{p,s} - \rho^j_p \right) = k_p A_i M_j \left(c^j_{p,s} - c^j_p \right) \tag{14.643}$$

Under the dynamic equilibrium condition on the phase interface the overall mass balance must be satisfied:

$$\dot{m}_{q^i} + \dot{m}_{p^j} = 0 \tag{14.644}$$

At the interface, the equilibrium relation given in Equation 14.650 (p. 774) also applies:

$$\rho^i_{q,s} = K^{\rho}_{q^i p^j} \rho^j_{p,s} \tag{14.645}$$

where $K^{\rho}_{q^i p^j}$ are the equilibrium ratios for mass concentration.

From Equation 14.642 (p. 772) — Equation 14.645 (p. 773), one can determine the interface mass concentrations:

$$\frac{\rho^i_{q,s}}{K^{\rho}_{q^i p^j}} = \rho^j_{p,s} = \frac{k_q \rho^i_q + k_p \rho^j_p}{K^{\rho}_{q^i p^j} k_q + k_p} \tag{14.646}$$

and then obtain the interface mass transfer rates:

$$\dot{m}_{p^i q^j} = \dot{m}_{q^j} = -\dot{m}_{p^i} = k_{pq} A_i \left(K_{q^i p^j} \rho^j_p - \rho^i_q \right) \tag{14.647}$$

$$\frac{1}{k_{pq}} = \frac{1}{k_q} + \frac{K^{\rho}_{q^i p^j}}{k_p}$$

The phase-specific mass transfer coefficients, k_q and k_p, can each be determined from one of the methods described in Mass Transfer Coefficient Models (p. 776). It is also possible to specify a zero-resistance condition on one side of the interface. This is equivalent to an infinite phase-specific mass transfer coefficient. For example, if $k_q \to \infty$ its effect is to force the interface concentration to be the same as the bulk concentration in phase q. In addition, if the overall mass transfer coefficient is known, k_{pq} can be specified directly by a constant or a user-defined function.

14.7.7.2. Equilibrium Models

Equilibrium models are used to compute equilibrium ratios $K_{q^i p^j}$ in Equation 14.641 (p. 771). The equilibrium models consider the case in which the species on the two phases are in dynamic equilibrium. The interphase mass transfer rate is determined from the relationship between the equilibrium species concentrations on the two phases.

Typically at equilibrium the species concentrations on the two phases are not the same. However, there exists a well-defined equilibrium curve relating the two concentrations. For binary mixtures, the equilibrium curve depends on the temperature and pressure. For multi-component mixtures, it is also a function of the mixture composition. The equilibrium curve is usually monotonic and nonlinear, and is often expressed in terms of equilibrium mole fractions of species i and j on phases q and p:

$$X^i_{q,e} = F\left(X^j_{p,e} \right) \tag{14.648}$$

The simplest curve or relationship is quasi-linear and assumes that at equilibrium the mole fractions of the species between the phases are in proportion:

$$X^i_{q,e} = K^x_{q^i p^j} X^j_{p,e} \tag{14.649}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

where $K^{x}_{q^i p^j}$ is the mole fraction equilibrium ratio. This relationship can, alternatively, be expressed in terms of c^i_q, ρ^i_q, or Y^i_q:

$$\begin{aligned} c^i_{q,e} &= K^{c}_{q^i p^j} c^j_{p,e} \\ \rho^i_{q,e} &= K^{\rho}_{q^i p^j} \rho^j_{p,e} \\ Y^i_{q,e} &= K^{y}_{q^i p^j} Y^j_{p,e} \end{aligned} \tag{14.650}$$

where $K^{c}_{q^i p^j}$, $K^{\rho}_{q^i p^j}$, and $K^{y}_{q^i p^j}$ are the equilibrium ratios for molar concentration, mass concentration, and mass fraction, respectively. These equilibrium ratios are related by the following expression:

$$K^{\rho}_{q^i p^j} = K^{c}_{q^i p^j} = \frac{c_q}{c_p} K^{x}_{q^i p^j} = \frac{\rho_q}{\rho_p} K^{y}_{q^i p^j} \tag{14.651}$$

There are several well-known interphase equilibrium models to define or determine $K^{x}_{q^i p^j}$ for various physical processes. Ansys Fluent offers three ways to determine $K^{x}_{q^i p^j}$:

- Raoult's Law (p. 774)
- Henry's Law (p. 775)
- Equilibrium Ratio (p. 776)

To quantify the concentration of a species, i, in phase q, several different, but related, variables are used:

c^i_q = Molar concentration of species i in phase q

ρ^i_q = Mass concentration of species i in phase q

X^i_q = Molar fraction of species i in phase q

Y^i_q = Mass fraction of species i in phase q

These four quantities are related as follows:

$$\rho^i_q = M^i c^i_q$$

$$X^i_q = \frac{c^i_q}{c_q}$$

$$Y^i_q = \frac{\rho^i_q}{\rho_q}$$

where c_q is the sum of the molar concentrations of all species in phase q, and ρ_q is the phase density of phase q.

14.7.7.2.1. Raoult's Law

In gas-liquid systems, the equilibrium relations are most conveniently expressed in terms of the partial pressure of the species in the gas phase. It is well known that for a pure liquid in contact with a gas mixture containing its vapor dynamic equilibrium occurs when the partial pressure of the vapor species is equal to its saturation pressure (a function of temperature). Raoult's law extends this statement to an ideal liquid mixture in contact with a gas. It states that the partial

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

pressure of the vapor species in phase p, P_p^j, is equal to the product of its saturation pressure, P_{sat}^j, and the molar fraction in the liquid phase, $X_{q,e}^i$:

$$P_p^j = P_{sat}^j X_{q,e}^i \tag{14.652}$$

If the gas phase behaves as an ideal gas, then Dalton's law of partial pressure gives:

$$P_p^j = X_{p,e}^j P \tag{14.653}$$

Using Equation 14.649 (p. 773), Equation 14.652 (p. 775), and Equation 14.653 (p. 775), Raoult's law can be rewritten in terms of a molar fraction ratio:

$$X_{q,e}^i = \frac{P}{P_{sat}^j} X_{p,e}^j \tag{14.654}$$

where the equilibrium ratio, $K_{q^i p^j}^x = \frac{P}{P_{sat}^j}$. While Raoult's law represents the simplest form of the vapor-liquid equilibrium (VLE) equation, keep in mind that it is of limited use, as the assumptions made for its derivation are usually unrealistic. The most critical assumption is that the liquid phase is an ideal solution. This is not likely to be valid, unless the system is made up of species of similar molecular sizes and chemical nature, such as in the case of benzene and toluene, or n-heptane and n-hexane (see Vapor Liquid Equilibrium Theory (p. 488)).

14.7.7.2.2. Henry's Law

Raoult's Law applies only for a phase with an ideal liquid mixture. To handle the case of a gas species dissolved into a non-ideal liquid phase, Henry's law provides a more generalized equilibrium relationship by replacing the saturation pressure, P_{sat}^j, in Equation 14.652 (p. 775) with a constant, H^x, for the liquid species i, referred to as Henry's constant:

$$P_p^j = H^x X_{q,e}^i \tag{14.655}$$

H^x has units of pressure and is known empirically for a wide range of materials, in particular for common gases dissolved in water. It is usually strongly dependent on temperature. The above form of Henry's laws is typically valid for a dilute mixture (mole fraction less than 0.1) and low pressures (less than 20 bar) [599] (p. 1087).

As with Raoult's law, Henry's law can be combined with Dalton's law to yield an expression in terms of equilibrium ratio:

$$X_{q,e}^i = K_{q^i p^j}^x X_{p,e}^j \tag{14.656}$$

where the equilibrium ratio is $K_{q^i p^j}^x = \frac{P}{H^x}$.

In addition to the molar fraction Henry's constant, H^x, the molar concentration Henry's constant, H^c is also commonly used:

$$P_p^j = H^c c_{q,e}^i \tag{14.657}$$

Fluent offers three methods for determining the Henry's constants: constant, the Van't Hoff correlation, and user-defined. In the Van't Hoff correlation, the Henry's constant is a function of temperature:

$$K_H = K_H^0 \exp\left(-\frac{\Delta_{soln} H}{R}\left(\frac{1}{T} - \frac{1}{T_{ref}}\right)\right) \tag{14.658}$$

and

$$H^c = \frac{1}{K_H} \quad H^x = \left(\frac{c^i_{q,e}}{X^i_{q,e}}\right)\frac{1}{K_H} \tag{14.659}$$

where $\Delta_{soln}H$ is the enthalpy of solution and K^0_H is the Henry's constant at the reference temperature, T_{ref}. The temperature dependence is:

$$-\frac{d\ln K_H}{d\left(1/T\right)} = \frac{\Delta_{soln}H}{R} \tag{14.660}$$

When using the Van't Hoff correlation in Fluent you specify the value of the reference Henry's constant, K^0_H, and the temperature dependence defined in Equation 14.660 (p. 776). These are material-dependent constants and can be found in [566] (p. 1086). By default, the constants for oxygen are used. You can also choose user-defined constants.

Important:

Regardless of the units selected in the Fluent user interface, K_H is always expressed in units of M/atm for consistency with the presentation in most reference data tables. Appropriate unit conversions are applied inside Fluent.

14.7.7.2.3. Equilibrium Ratio

In situations where neither Raoult's law nor Henry's law apply, such as liquid-liquid extraction, or multiple species mass transfers, Fluent offers you the option to directly specify the equilibrium ratios. You may choose to specify the equilibrium ratio in the following forms:

Molar concentration equilibrium ratio, $K^c_{q^i p^j}$

Molar fraction equilibrium ratio, $K^x_{q^i p^j}$

Mass fraction equilibrium ratio, $K^y_{q^i p^j}$

These specify the ratio of the concentration variable for the **From**-phase divided by that for the **To**-phase as defined in Equation 14.649 (p. 773) and Equation 14.650 (p. 774). You can specify either a constant value or a user-defined function.

14.7.7.3. Mass Transfer Coefficient Models

In Fluent, the overall mass transfer coefficient (k_{pq} in Equation 14.647 (p. 773)) can be modeled as a constant or a user-defined function (UDF). The phase mass transfer coefficient (for instance, k_q in Equation 14.647 (p. 773)) is defined on either side of the phase interface and can be modeled as a constant, a UDF, or a function of the phase Sherwood number. In a phase q, the Sherwood number Sh_q is expressed as:

$$k_q = \frac{\mathrm{Sh}_q D_q}{L_q} \tag{14.661}$$

where D_q is the diffusivity of the qth phase, and L_q is a characteristic length (such as a bubble or droplet diameter).

The mass transfer coefficient in phase p, k_p, is defined similarly to k_q in Equation 14.661 (p. 776).

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

The Sherwood number is either specified as a constant or determined from one of several empirical correlations. These methods are detailed in the following sections:

14.7.7.3.1. Constant
14.7.7.3.2. Sherwood Number
14.7.7.3.3. Ranz-Marshall Model
14.7.7.3.4. Hughmark Model
14.7.7.3.5. Higbie Model
14.7.7.3.6. User-Defined

14.7.7.3.1. Constant

When modeled as a constant, the constant volumetric mass transfer coefficient (k_{pq}, k_p, or k_q) is specified by the user.

14.7.7.3.2. Sherwood Number

In the Sherwood number method, the user specifies a constant value for Sh_q, and the phase mass transfer coefficient, k_q, is computed from Equation 14.661 (p. 776).

14.7.7.3.3. Ranz-Marshall Model

The Ranz-Marshall model uses an analogous approach to that for the Ranz-Marshall heat transfer coefficient model. The expression for the Sherwood number for flow past a spherical particle has the same form as that for the Nusselt number in the context of heat transfer, with the Prandtl number replaced by the Schmidt number:

$$\mathrm{Sh}_q = 2 + 0.6\mathrm{Re}_q^{1/2}\mathrm{Sc}_q^{1/3} \tag{14.662}$$

where Sc_q is the Schmidt number of the q$^{\text{th}}$ phase and Re_q is the relative Reynolds number:

$$\mathrm{Sc}_q = \frac{\mu_q}{\rho_q D_q} \qquad \mathrm{Re}_q = \frac{\rho_q \left|\vec{V}_q - \vec{V}_p\right| d_p}{\mu_q}$$

where μ_q and ρ_q are, respectively, the dynamic viscosity and density of the q$^{\text{th}}$ phase. $\left|\vec{V}_q - \vec{V}_p\right|$ is the magnitude of the relative velocity between phases. The Ranz-Marshall model is based on boundary layer theory for steady flow past a spherical particle. It generally applies under the conditions:

$$0 \le \mathrm{Re}_q < 200$$
$$0 \le \mathrm{Sc}_q < 250$$

14.7.7.3.4. Hughmark Model

Like the Ranz-Marshall model, the Hughmark model [261] (p. 1068) for mass transfer coefficient is also analogous to its heat transfer coefficient counterpart. The Ranz-Marshall model is extended to a wider range of Reynolds numbers by the Hughmark correction:

$$\mathrm{Sh}_q = \begin{cases} 2 + 0.6\mathrm{Re}_q^{1/2}\mathrm{Sc}_q^{1/3} & 0 \le \mathrm{Re}_q < 776.06 \quad 0 \le \mathrm{Sc}_q < 250 \\ 2 + 0.27\mathrm{Re}_q^{0.62}\mathrm{Sc}_q^{1/3} & 776.06 \le \mathrm{Re}_q \quad 0 \le \mathrm{Sc}_q < 250 \end{cases} \tag{14.663}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

The Reynolds number crossover point is chosen to ensure continuity. The model should not be used outside of the recommended Schmidt number range.

14.7.7.3.5. Higbie Model

The Higbie mass transfer coefficient model provides a mass transfer coefficient for interphase species mass transfer based on Higbie's penetration theory [196] (p. 1064). The Higbie model uses the following correlation:

$$k_q = \frac{2}{\sqrt{\pi}} \sqrt{D_L} \left(\frac{\varepsilon \rho}{\mu} \right)^{\frac{1}{4}} \tag{14.664}$$

where

D_L = liquid phase species diffusion coefficient

ε, ρ, μ = turbulent eddy dissipation, density, and viscosity of the liquid phase, respectively

14.7.7.3.6. User-Defined

You can also use a user defined function for the mass transfer coefficient. To create a user-defined function, use the `DEFINE_MASS_TRANSFER` macro as you would in other mass transfer cases.

14.8. Modeling Species Transport in Multiphase Flows

Species transport, as described in Species Transport and Finite-Rate Chemistry (p. 237), can also be applied to multiphase flows. You can choose to solve the conservation equations for chemical species in multiphase flows by having Ansys Fluent, for each phase k, predict the local mass fraction of each species, Y_i^k, through the solution of a convection-diffusion equation for the i^{th} species. The generalized chemical species conservation equation (Equation 7.1 (p. 238)), when applied to a multiphase mixture can be represented in the following form:

$$\frac{\partial}{\partial t}\left(\rho^q \alpha^q Y_{iq}\right) + \nabla \cdot \left(\rho^q \alpha^q \vec{v}^q Y_{iq}\right) = -\nabla \cdot \alpha^q \vec{J}_{iq} + \alpha^q R_{iq} + \alpha^q S_{iq} + \sum_{p=1}^{n} \dot{m}_{p^i q^j} + \mathcal{R} \tag{14.665}$$

where R_i^q is the net rate of production of homogeneous species i by chemical reaction for phase q, $\dot{m}_{q^j p^i}$ is the mass transfer source between species i and j from phase q to p, and $\mathfrak{R}$ is the heterogeneous reaction rate. In addition, α^q is the volume fraction for phase q and S_i^q is the rate of creation by addition from the dispersed phase plus any user-defined sources.

Ansys Fluent treats homogeneous gas phase chemical reactions the same as a single-phase chemical reaction. The reactants and the products belong to the same mixture material (set in the **Species Model** dialog box), and hence the same phase. The reaction rate is scaled by the volume fraction of the particular phase in the cell.

The set-up of a homogeneous gas phase chemical reaction in Ansys Fluent is the same as it is for a single phase. For more information, see Species Transport and Finite-Rate Chemistry (p. 237). For most multiphase species transport problems, boundary conditions for a particular species are set in the associated phase boundary condition dialog box (see Defining Multiphase Cell Zone and Boundary Conditions

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

in the User's Guide), and postprocessing and reporting of results is performed on a per-phase basis (see Postprocessing for Multiphase Modeling in the User's Guide).

For multiphase species transport simulations, the **Species Model** dialog box allows you to include **Volumetric**, **Wall Surface**, and **Particle Surface** reactions. Ansys Fluent treats multiphase surface reactions as it would a single-phase reaction. The reaction rate is scaled with the volume fraction of the particular phase in the cell. For more information, see Species Transport and Finite-Rate Chemistry (p. 237).

Important:

To turn off reactions for a particular phase, while keeping the reactions active for other phases, turn on **Volumetric** under **Reactions** in the **Species Model** dialog box. Then, in the **Create/Edit Materials** dialog box, select **none** from the **Reactions** drop-down list.

The species of different phases is entirely independent. There is no implicit relationship between them even if they share the same name. Explicit relationships between species of different phases can be specified through mass transfer and heterogeneous reactions. For more information on mass transfer and heterogeneous reactions, see Including Mass Transfer Effects and Specifying Heterogeneous Reactions in the User's Guide, respectively.

Some phases may have a fluid material associated with them instead of a mixture material. The species equations are solved in those phases that are assigned a mixture material. The species equation above is solved for the mass fraction of the species in a particular phase. The mass transfer and heterogeneous reactions will be associated with the bulk fluid for phases with a single fluid material.

Additional information about modeling species transport is presented in the following subsections:

14.8.1. Limitations

14.8.2. Mass and Momentum Transfer with Multiphase Species Transport

14.8.3. The Stiff Chemistry Solver

14.8.4. Heterogeneous Phase Interaction

14.8.1. Limitations

The following limitations exist for the modeling of species transport for multiphase flows:

- The nonpremixed, premixed, partially-premixed combustion, or the composition PDF transport species transport models are not available for multiphase species reactions.
- Only the laminar finite-rate, finite-rate/eddy-dissipation, and eddy-dissipation turbulence-chemistry models of homogeneous reactions are available for multiphase species transport.
- The discrete phase model (DPM) is not compatible with multiphase species transport.

14.8.2. Mass and Momentum Transfer with Multiphase Species Transport

The Ansys Fluent multiphase mass transfer model accommodates mass transfer between species belonging to different phases. Instead of a matrix-type input, multiple mass transfer mechanisms need to be input. Each mass transfer mechanism defines the mass transfer phenomenon from one entity to another entity. An entity is either a particular species in a phase, or the bulk phase itself if the phase does not have a mixture material associated with it. The mass transfer phenomenon could be

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

specified either through the built-in unidirectional "constant-rate" mass transfer (Unidirectional Constant Rate Mass Transfer (p. 752)) or through user-defined functions.

Ansys Fluent loops through all the mass transfer mechanisms to compute the net mass source/sink of each species in each phase. The net mass source/sink of a species is used to compute species and mass source terms. Ansys Fluent will also automatically add the source contribution to all relevant momentum and energy equations based on that assumption that the momentum and energy carried along with the transferred mass. For other equations, the transport due to mass transfer must be explicitly modeled by you.

14.8.2.1. Source Terms Due to Heterogeneous Reactions

Consider the following reaction:

$$aA+bB \rightarrow cC+dD \tag{14.666}$$

Let as assume that A and C belong to phase 1 and B and D to phase 2.

14.8.2.1.1. Mass Transfer

Mass source for the phases are given by:

$$S_1 = \mathfrak{R}\left(cM_c - aM_a\right) \tag{14.667}$$

$$S_2 = \mathfrak{R}\left(dM_d - bM_b\right) \tag{14.668}$$

where S is the mass source, M is the molecular weight, and $\mathfrak{R}$ is the reaction rate.

The general expression for the mass source for the i^{th} phase is

$$S_{r_i} = -\mathfrak{R}\sum_{r_i}\gamma_j^r M_j^r \tag{14.669}$$

$$S_{p_i} = \mathfrak{R}\sum_{p_i}\gamma_j^p M_j^p \tag{14.670}$$

$$S_i = S_{p_i} + S_{r_i} \tag{14.671}$$

where γ is the stoichiometric coefficient, p represents the product, and r represents the reactant.

14.8.2.1.2. Momentum Transfer

Momentum transfer is more complicated, but we can assume that the reactants mix (conserving momentum) and the products take momentum in the ratio of the rate of their formation.

The net velocity, $\vec{u}_{net}$, of the reactants is given by:

$$\vec{u}_{net} = \frac{aM_a\vec{u}_1 + bM_b\vec{u}_2}{aM_a + bM_b} \tag{14.672}$$

The general expression for the net velocity of the reactants is given by:

$$\vec{u}_{net} = \frac{\sum_r \gamma_j^r M_j^r \vec{u}_{r_j}}{\sum_r \gamma_j^r M_j^r} \tag{14.673}$$

where j represents the j^{th} item (either a reactant or a product).

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Momentum transfer for the phases is then given by:

$$S_1^{\vec{u}} = \mathcal{R}\left(cM_c\vec{u}_{net} - aM_a\vec{u}_1\right) \tag{14.674}$$

$$S_2^{\vec{u}} = \mathcal{R}\left(dM_d\vec{u}_{net} - bM_b\vec{u}_2\right) \tag{14.675}$$

The general expression is

$$S_i^{\vec{u}} = S_{p_i}\vec{u}_{net} - \mathcal{R}\sum_{r_i}\gamma_j^r M_j^r\vec{u}_i \tag{14.676}$$

If we assume that there is no momentum transfer, then the above term will be zero.

14.8.2.1.3. Species Transfer

The general expression for source for k^{th} species in the j^{th} phase is

$$S_{r_i^k} = -\mathcal{R}\sum_{r_i^k}\gamma_j^{r^k} M_j^{r^k} \tag{14.677}$$

$$S_{p_i^k} = \mathcal{R}\sum_{p_i^k}\gamma_j^{p^k} M_j^{p^k} \tag{14.678}$$

$$S_i^k = S_{p_i^k} + S_{r_i^k} \tag{14.679}$$

14.8.2.1.4. Heat Transfer

For heat transfer, we need to consider the formation enthalpies of the reactants and products as well:

The net enthalpy of the reactants is given by:

$$H_{net} = \frac{aM_a\left(H_a + h_a^f\right) + bM_b\left(H_b + h_b^f\right)}{aM_a + bM_b} \tag{14.680}$$

where h^f represents the formation enthalpy, and H represents the enthalpy.

The general expression for H_{net} is:

$$H_{net} = \frac{\sum_r \gamma_j^r M_j^r\left(H_j^r + h^{f^r}{}_j\right)}{\sum_r \gamma_j^r M_j^r} \tag{14.681}$$

If we assume that this enthalpy gets distributed to the products in the ratio of their mass production rates, heat transfer for the phases are given by:

$$S_1^H = \mathcal{R}\left(cM_cH_{net} - aM_aH^a - cM_ch_f^c\right) \tag{14.682}$$

$$S_2^H = \mathcal{R}\left(dM_dH_{net} - bM_bH^b - dM_dh_f^d\right) \tag{14.683}$$

The last term in the above equations appears because our enthalpy is with reference to the formation enthalpy.

The general expression for the heat source is:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$S_i^H = S_{p_i} H_{net} - \mathfrak{R}\left(\sum_{r_i} \gamma_j^r M_j^r H_j^r + \sum_{p_i} \gamma_j^p M_j^p h^{f^p}{}_j \right) \tag{14.684}$$

If we assume that there is no heat transfer, we can assume that the different species only carry their formation enthalpies with them. Therefore the expression for H_{net} will be:

$$H_{net} = \frac{\sum_r \gamma_j^r M_j^r h^{f^r}{}_j}{\sum_r \gamma_j^r M_j^r} \tag{14.685}$$

The expression S_i^H will be

$$S_i^H = S_{p_i} H_{net} - \mathfrak{R} \sum_{p_i} \gamma_j^p M_j^p h^{f^p}{}_j \tag{14.686}$$

14.8.3. The Stiff Chemistry Solver

In Ansys Fluent, the stiff chemistry solver can solve the following chemical reactions:

- Interphase (or heterogenous) reactions

  This is only available for unsteady cases where a fractional step scheme is applied. In the first fractional step, the multiphase species Equation 14.665 (p. 778) is solved spatially with the reaction term R_i^q set to zero. In the second fractional step, the reaction term is integrated in every cell using a stiff ODE solver. To use this option, refer to Specifying Heterogeneous Reactions in the User's Guide.

- Intraphase (or homogenous) reactions

  In this case, the solution algorithm is similar to that used in single-phase reacting flows described in Volumetric Reactions (p. 237). ISAT can be used to accelerate detailed stiff chemistry (see In-Situ Adaptive Tabulation (ISAT) (p. 284) for more information). This is only available for Eulerian multiphase cases, where the stiff chemistry solver can be used for only one of the mixture phases.

14.8.4. Heterogeneous Phase Interaction

To compute the heterogeneous phase interaction rates, a modified Arrhenius type rate expression is provided, which you can define using the graphical user interface (see Specifying Heterogeneous Reactions in the User's Guide), or a user-defined function (see `DEFINE_HET_RXN_RATE` in the Fluent Customization Manual).

The rate expression is in the general form:

$$Rate = k \Pi_{i=1}^{NR} \left(\frac{Y_{ip} \rho_p \alpha_p}{MW_{\text{ip}}} \right)^n \; kmol/\, m^3\text{-}sec \tag{14.687}$$

where

$$k = A \left(\frac{T_p}{T_{ref}} \right)^b exp\left(\frac{-E}{RT_p} \right) \tag{14.688}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Here, Y_{ip} is the mass fraction of reactant species i in the phase p, NR is the total number of reactants in a given inter-phase reaction, ρ_p is the bulk density of the phase p, α_p is the volume fraction of the phase p, MW_{ip} the molecular weight (kg/kmol) of the reactant species i in the phase p, n is the rate exponent, and k is the rate constant.

The rate constant in the modified Arrhenius form is given by Equation 14.688 (p. 782). T_p is the phase temperature (phase-1, phase-2, and so on) as required by the expression. Make sure you provide the correct phase for the temperature to be extracted. T_{ref} is the normalization temperature. Usually, T_{ref} is set to unity for most reactions, but certain reaction rate constants may have a given T_{ref} value, which is normally 298.15 K.

The reaction rate given in Equation 14.688 (p. 782) will be valid above the kick-off temperature specified by you, where the temperature for the rate calculation is taken from the selected phase.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information
of ANSYS, Inc. and its subsidiaries and affiliates.

Chapter 15: Population Balance Model

Several industrial fluid flow applications involve a secondary phase with a size distribution. The size distribution of particles, including solid particles, bubbles, or droplets, can evolve in conjunction with transport and chemical reaction in a multiphase system. The evolutionary processes can be a combination of different phenomena like nucleation, growth, dispersion, dissolution, aggregation, and breakage producing the dispersion. Thus in multiphase flows involving a size distribution, a balance equation is required to describe the changes in the particle population, in addition to momentum, mass, and energy balances. This balance is generally referred to as the population balance. Cases in which a population balance could apply include crystallization, precipitative reactions from a gas or liquid phase, bubble columns, gas sparging, sprays, fluidized bed polymerization, granulation, liquid-liquid emulsion and separation, and aerosol flows.

This chapter describes the theory behind the population balance model available in Ansys Fluent. Information in this chapter is organized into the following sections:

15.1. Introduction

To make use of this modeling concept, a number density function is introduced to account for the particle population. With the aid of particle properties (for example, particle size, porosity, composition, and so on), different particles in the population can be distinguished and their behavior can be described.

Ansys Fluent offers three solution methods to the population balance equation: discretized population balance, standard method of moments, and quadrature method of moments.

15.1.1. The Discrete Method

15.1.2. The Inhomogeneous Discrete Method

15.1.3. The Standard Method of Moments

15.1.4. The Quadrature Method of Moments

15.1.1. The Discrete Method

In the discrete method, the particle population is discretized into a finite number of size intervals. This approach has the advantage of computing the particle size distribution (PSD) directly. This approach is also particularly useful when the range of particle sizes is known *a priori* and does not span more than two or three orders of magnitude. In this case, the population can be discretized with a relatively small number of size intervals and the size distribution that is coupled with fluid dynamics can be computed. The disadvantage of the discrete method is that it is computationally expensive if a large number of intervals is needed.

15.1.2. The Inhomogeneous Discrete Method

One of the limitations of the existing homogeneous discrete method is that all bins are assigned to the same secondary phase and are therefore advected with the same phase momentum. This is un-

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

suitable for modeling cases where large and small bin sizes are likely to segregate due to different momentum fields. The inhomogeneous discrete method overcomes this limitation by allowing groups of bins to be advected by different phase velocities. Therefore when the inhomogeneous discrete model is activated, the Population Balance model can be applied to more than one secondary phase.

The general transport equation for the discrete bin fraction f_i can be written as

$$\frac{\partial}{\partial t}(\rho\alpha f_i) + \nabla\cdot(\vec{u_p}\alpha f_i) = S_{bi} \tag{15.1}$$

Since all bins belong to a single phase in the homogeneous discrete method, the net mass source for the phase in case of breakage and agglomeration is zero and can be expressed as

$$\sum_{i=1}^{M} S_{bi} = 0 \tag{15.2}$$

This is shown schematically in Figure 15.1: Homogeneous Discrete Method (p. 787) where all bins are advected by the same phase velocity u_p. In contrast, the inhomogeneous discrete method shown in Figure 15.2: Inhomogeneous Discrete Method (p. 787) allows bins to be assigned to multiple phases. Here M bins per phase are distributed over N phases for a total of $M \times N$ bins. Bins f_1 and f_M are advected by phase velocity u_{p_1} and so forth. The sum of bin sources in any given phase is not necessarily equal to zero since bins in a given phase can migrate to another phase through breakage or agglomeration, thereby creating a net mass source for that phase.

The net mass source for a given phase can be expressed as the sum of the bin sources belonging to that phase

$$S_i = \sum_{i=1}^{M} S_{bi} \tag{15.3}$$

For breakage and coalescence the sum over all phase sources is zero

$$\sum_{i=1}^{N} S_i = 0 \tag{15.4}$$

also similar to the homogeneous discrete model

$$\sum_{i=1}^{M} f_i = 1 \tag{15.5}$$

Important:

The inhomogeneous discrete method is currently limited to breakage and coalescence only.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Figure 15.1: Homogeneous Discrete Method

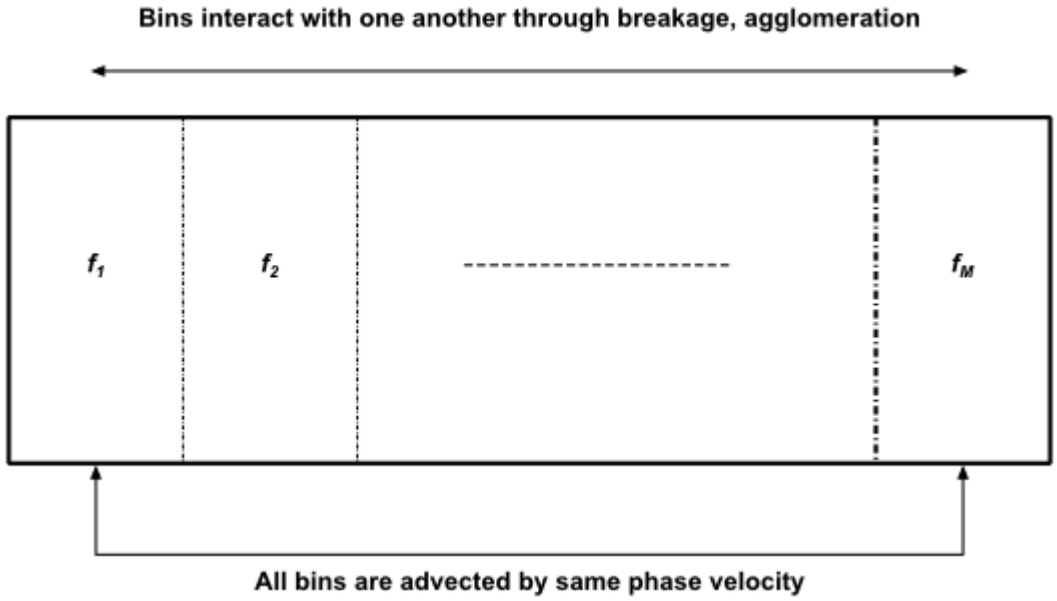
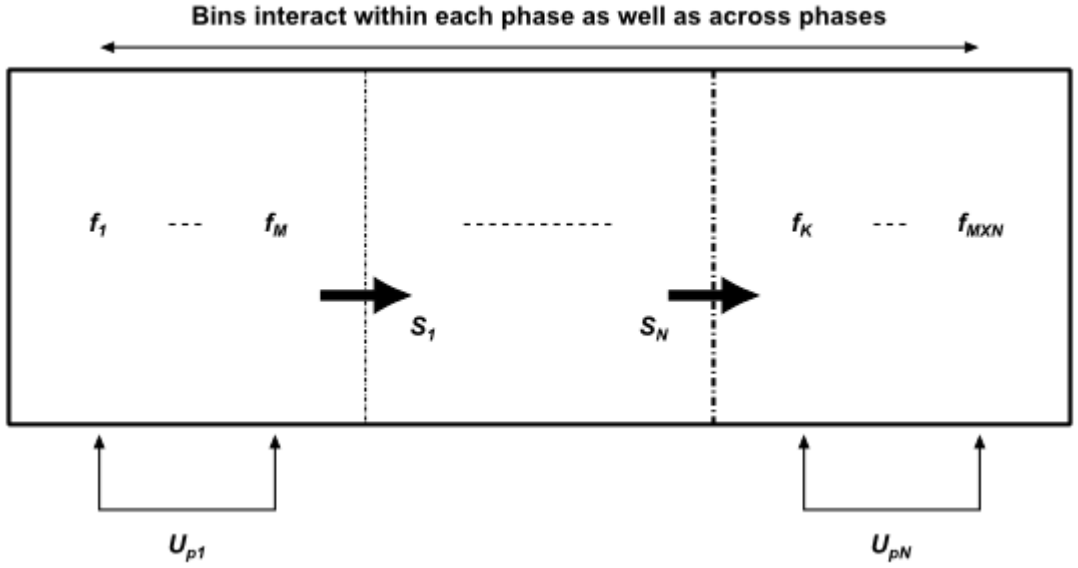


Figure 15.2: Inhomogeneous Discrete Method



15.1.3. The Standard Method of Moments

The standard method of moments (SMM) is an efficient alternative to the discrete population balance approach. In this approach, the population balance equation is transformed into a set of transport equations for moments of the distribution. The i^{th} moment is defined by integrating the number density throughout the particle space weighted with the particle property raised to its i^{th} power. It is generally sufficient to solve only a few moment equations, typically three to six. This may provide a significant reduction in the number of equations to be solved compared with the discretized approach. Apart from the computational advantage, the SMM approach is useful when the entire distribution is not needed and certain average and total quantities are sufficient to represent the particle distribution. Typically, the zeroth moment represents the total number density, the second moment

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

represents the total surface area per unit volume, and the third moment represents the total mass density.

In the SMM approach, no assumptions are made about the size distribution, and the moment equations are formulated in a closed form involving only functions of the moments themselves. However, this exact closure requirement poses a serious limitation, as aggregation (with the exception of the constant aggregation kernel) and breakage phenomena cannot be written as functions of moments.

15.1.4. The Quadrature Method of Moments

The quadrature method of moments (QMOM) has a similar advantage as the SMM in terms of computational costs, but replaces the exact closure needed by SMM with an approximate closure. This allows application of QMOM to a broad range of applications without any limitations.

15.2. Population Balance Model Theory

This chapter presents an overview of the theory and the governing equations for the methods used in Ansys Fluent to predict particle growth and nucleation.

15.2.1. The Particle State Vector

15.2.2. The Population Balance Equation (PBE)

15.2.3. Solution Methods

15.2.4. Population Balance Statistics

15.2.1. The Particle State Vector

The particle state vector is characterized by a set of "external coordinates" ($\vec{x}$), which denote the spatial position of the particle, and "internal coordinates" (ϕ), which could include particle size, composition, and temperature. From these coordinates, a number density function $n\left(\vec{x},\phi,t\right)$ can be postulated where $\phi \in \Omega_{\phi}$, $\vec{x} \in \Omega_{\vec{x}}$. Therefore, the average number of particles in the infinitesimal volume $dV_{\vec{x}}dV_{\phi}$ is $n\left(\vec{x},\phi,t\right)dV_{\vec{x}}dV_{\phi}$. In contrast, the continuous phase state vector is given by $\vec{Y} \equiv \left[Y_1\left(\vec{x},t\right), Y_2\left(\vec{x},t\right), \ldots, Y_C\left(\vec{x},t\right)\right]$

The total number of particles in the entire system is then

$$\int_{\Omega_{\phi}}\int_{\Omega_{\vec{x}}} n dV_{\vec{x}} dV_{\phi} \tag{15.6}$$

The local average number density in physical space (that is, the total number of particles per unit volume) is given by

$$N\left(\vec{x},t\right) = \int_{\Omega_{\phi}} n dV_{\phi} \tag{15.7}$$

The total volume fraction of all particles is given by

$$\alpha\left(\vec{x},t\right) = \int_{\Omega_{\phi}} nV\left(\phi\right) dV_{\phi} \tag{15.8}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

where $V(\phi)$ is the volume of a particle in state ϕ.

15.2.2. The Population Balance Equation (PBE)

Assuming that ϕ is the particle volume, the transport equation for the number density function is given as

$$\frac{\partial}{\partial t}\left[n(V,t)\right] + \nabla\cdot\left[\vec{u}n(V,t)\right] + \underbrace{\nabla_v\cdot\left[G_v n(V,t)\right]}_{\text{Growth term}} = \underbrace{\frac{1}{2}\int_0^V a(V-V',V')n(V-V',t)n(V',t)dV'}_{\text{Birth due to aggregation}} - \underbrace{\int_0^\infty a(V,V')n(V,t)n(V',t)dV'}_{\text{Death due to aggregation}} + \underbrace{\int_{\Omega_v} pg(V')\beta(V|V')n(V',t)dV'}_{\text{Birth due to breakage}} - \underbrace{g(V)n(V,t)}_{\text{Death due to breakage}} \tag{15.9}$$

The boundary and initial conditions are given by

$$n(V,t=0)=n_v;\quad n(V=0,t)G_v=\dot{n}_0 \tag{15.10}$$

where $\dot{n}_0$ is the nucleation rate in particles/m^3-s.

15.2.2.1. Particle Growth

The growth rate based on particle volume, G_v, (m^3/s) is defined as

$$G_v=\frac{\partial V}{\partial t} \tag{15.11}$$

The growth rate based on particle diameter (or length) is defined as

$$G=\frac{\partial L}{\partial t} \tag{15.12}$$

The volume of a single particle V is defined as $K_v L^3$, and therefore the relationship between G_v and G is

$$G_v=3K_v L^2 G \tag{15.13}$$

The surface area of a single particle, A, is defined as $K_a L^2$. Thus for a cube or a sphere, $K_a=6K_v$.

Important:

- In the case of dissolution of particles, the particle volume growth rate is negative due to the particle volume reduction.
- In the case of bubble expansion due to change in density (pressure), the growth rate G_v in Equation 15.9 (p. 789) is the rate of bubble volume expansion.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Considering m to be the mass of a bubble of volume V and density ρ,

$$\begin{aligned} m &= \rho V \\ \frac{dm}{dt} &= \rho \frac{dV}{dt} + V\frac{d\rho}{dt} \end{aligned} \tag{15.14}$$

For constant m (bubble expansion with no mass transfer), $dm/dt=0$, and therefore

$$G_V=-\frac{V}{\rho}\frac{d\rho}{dt} \tag{15.15}$$

A more detailed explanation can be found in the paper by Mills [440] (p. 1078).

15.2.2.2. Particle Birth and Death Due to Breakage and Aggregation

The birth and death of particles occur due to breakage and aggregation processes. Examples of breakage processes include crystal fracture in crystallizers and bubble breakage due to liquid turbulence in a bubble column. Similarly, aggregation can occur due to particle agglomeration in crystallizers and bubble coalescence in bubble column reactors.

Note that all breakage and aggregation events in FLUENT Population Balance Model are modeled as binary with the exception of the generalized breakage kernel.

15.2.2.2.1. Breakage

The breakage rate expression, or kernel [397] (p. 1076), is expressed as

$$\Omega_{br}=F_B g(V')\beta(V|V') \tag{15.16}$$

where

$g(V')$ = breakage frequency; that is, the fraction of particles of volume V' breaking per unit time (s^{-1})

$\beta(V|V')$ = probability density function (PDF) of particles breaking from volume V' to a particle of volume V ($1/m^3$)

F_B = breakage factor for the calibration of the breakage kernels

The birth rate of particles of volume V due to breakage is given by

$$B_{br}=\int_{\Omega_v} p g(V')\beta(V|V')n(V')dV' \tag{15.17}$$

where $g(V')n(V')dV'$ particles of volume V' break per unit time, producing $pg(V')n(V')dV'$ particles, of which a fraction $\beta(V|V')dV$ represents particles of volume V. p is the number of child particles produced per parent (for example, two particles for binary breakage).

The death rate of particles of volume V due to breakage is given by

$$D_{br}=g(V)n(V) \tag{15.18}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

The PDF $\beta\left(V\middle|V'\right)$ is also known as the particle fragmentation distribution function, or daughter size distribution. Several functional forms of the fragmentation distribution function have been proposed, though the following physical constraints must be met: the normalized number of breaking particles must sum to unity, the masses of the fragments must sum to the original particle mass, and the number of fragments formed has to be correctly represented.

Mathematically, these constraints can be written as follows:

- For the normalization condition:

$$\int_0^{V'} \beta\left(V\middle|V'\right)dV=1 \tag{15.19}$$

- For conservation of mass

$$p\int_0^{V'} m(V)\ \beta\left(V\middle|V'\right)dV=m\left(V'\right) \tag{15.20}$$

- For binary breakage, β is symmetric about $V/V'=0.5$; that is,

$$\beta\left(V'-V\middle|V'\right)=\beta\left(V\middle|V'\right) \tag{15.21}$$

The following is a list of models available in Ansys Fluent to calculate the breakage frequency:

- Constant value
- Luo model
- Lehr model
- Ghadiri model
- Laakkonen model
- Liao model
- User-defined model

Ansys Fluent provides the following models for calculating the breakage probability density functions (PDF):

- Parabolic PDF
- Laakkonen PDF
- Generalized PDF for multiple breakage fragments
- User-defined model

The breakage frequency models and the parabolic and generalized PDFs are described in detail in the sections that follow.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

15.2.2.2.2. Luo and Lehr Breakage Kernels

The Luo and Lehr models are integrated kernels, encompassing both the breakage frequency and the PDF of breaking particles.

The general form is the integral over the size of eddies λ hitting the particle with diameter d (and volume V). The integral is taken over the dimensionless eddy size $\xi = \lambda / d$. The general form is

$$\Omega_{br}(V, V') = K \int_{\xi_{min}}^{1} \frac{(1+\xi)^2}{\xi^n} exp(-b\xi^m)\, d\xi \tag{15.22}$$

where the parameters are as shown in Table 15.1: Luo Model Parameters (p. 792) and Table 15.2: Lehr Model Parameters (p. 792).

Table 15.1: Luo Model Parameters

K	n	b	m
$0.9238\varepsilon^{1/3}d^{-2/3}\alpha$	11/3	$12\left[f^{2/3}+(1-f)^{2/3}-1\right]\sigma\rho^{-1}\varepsilon^{-2/3}d^{-5/3}\beta^{-1}$ where $f = V/V'$	-11/3

For detailed notations, see Luo [397] (p. 1076).

Table 15.2: Lehr Model Parameters

K	n	b	m
$1.19\varepsilon^{-1/3}d^{-7/3}\sigma\rho^{-1}f^{-1/3}$	13/3	$2We_{crit}\sigma\rho^{-1}\varepsilon^{-2/3}d^{-5/3}f^{-1/3}$	-2/3

Where We_{crit} is entered through the GUI. See Lehr [351] (p. 1073).

15.2.2.2.3. Ghadiri Breakage Kernels

The Ghadiri model [201] (p. 1064), [457] (p. 1079), in contrast to the Luo and Lehr models, is used to model only the breakage frequency of the solid particles. You will have to specify the PDF model to define the daughter distribution.

The breakage frequency g is defined as:

$$g(V') = K_b v^2 d^{5/3} \tag{15.23}$$

where v is the particle velocity and d is the particle diameter prior to breaking. The breakage constant K_b [s^3 $m^{-11/3}$] sets the overall scale of how many parent particles is breaking per second. It is case-dependent and is proportional to the particle properties:

$$K_b \sim \frac{\rho E^{2/3}}{\Gamma^{5/3}} \tag{15.24}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

where ρ is the particle density, E is the elastic modulus of the granule, and Γ is the interface energy. Note that you may need to adjust K_b, for example, by calibrating against a baseline experiment.

15.2.2.2.4. Laakkonen Breakage Kernels

The Laakkonen breakage kernel is expressed as the product of a breakage frequency, $g(V')$ and a daughter PDF $\beta(V,V')$ where

$$g(V')=C_2\varepsilon^{1/3}erfc\left(\sqrt{C_3\frac{\sigma}{\rho_L\varepsilon^{2/3}d^{5/3}}+C_4\frac{\mu_L}{\sqrt{\rho_L\rho_G}\,\varepsilon^{1/3}d^{4/3}}}\right) \tag{15.25}$$

where ε is the liquid phase eddy dissipation, σ is the surface tension, ρ_l is the liquid density, ρ_g is the gas density, d is the parent particle diameter and μ_L is the liquid viscosity.

The constants C_2=2.52, C_3=0.04 and C_4=0.01.

The daughter PDF is given by

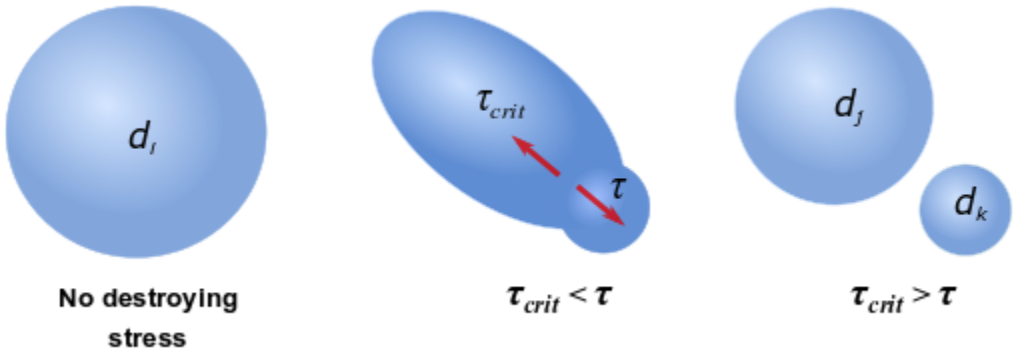
$$\beta(V,V')=\frac{30}{V'}\left(\frac{V}{V'}\right)^2\left(1-\frac{V}{V'}\right)^2 \tag{15.26}$$

where V and V' are the daughter and parent particle volumes, respectively. This model is a useful alternative to the widely used Luo model because it has a simple expression for the daughter PDF and therefore requires significantly less computational effort.

15.2.2.2.5. Liao Breakage Kernel

A bubble in a flowing liquid experiences a stress (τ) that deforms the bubble and eventually breaks it once the stress exceeds the restoring effect of the surface tension (τ_{crit}) as shown in Figure 15.3: Bubble Breakup (p. 793) (adapted from [366] (p. 1074)).

Figure 15.3: Bubble Breakup



Based on the work of Martínez-Bazán et al. [413] (p. 1077), Liao [366] (p. 1074) proposed to define the breakup frequency of a bubble of size d_i in relation to stress τ and critical stress τ_{crit} as follows:

$$\Omega_{br}(V, V') = \begin{cases} \frac{1}{d_i}\sqrt{\frac{\tau(d_i)-\tau_{crit}(d_i, d_j)}{\rho}} & for\ (\tau > \tau_{crit}) \\ 0 & for\ (\tau \le \tau_{crit}) \end{cases} \tag{15.27}$$

where d_j is the diameter of a child bubble.

The combined breakup approach is used to determine the critical stress based on consideration of energy criterion and force criterion. For binary breakage, where bubble i breaks into two bubbles j and k, τ_{crit} can be expressed as:

$$\tau_{crit} = max(\tau_{crit,1},\ \tau_{crit,2}) \tag{15.28}$$

The energy constraint $\tau_{crit,1}$ is determined by the increase of the surface energy during the breakage:

$$\tau_{crit,1} = E_{\sigma,j} + E_{\sigma,k} - E_{\sigma,i} = \frac{6\sigma}{d_i}\left(\left(\frac{d_j}{d_i}\right)^2 + \left(\frac{d_k}{d_i}\right)^2 - 1\right) \tag{15.29}$$

where E_σ is the surface energy of the bubble, and the size of bubble k is calculated based on the sizes of bubbles i and j as:

$$d_k = \left(d_i^3 + d_j^3\right)^{1/3} \tag{15.30}$$

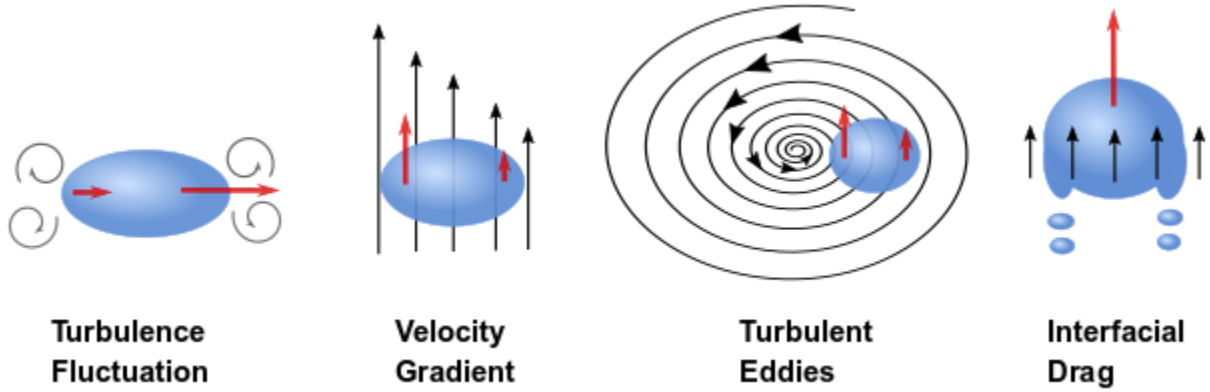
The force constraint $\tau_{crit,2}$ is determined by the capillary force of the smallest child bubble:

$$\tau_{crit,2} = \frac{\sigma}{min(d_k, d_j)} \tag{15.31}$$

where σ is the surface tension coefficient.

The destroying stress τ_{crit} that breaks the bubble is caused by the velocity gradient across the bubble's surface. In a turbulent flow, the velocity variations can result from various mechanisms such as turbulent fluctuation, velocity gradients in a mean flow, turbulent eddies, and interfacial drag [366] (p. 1074) as shown in Figure 15.4: Bubble Breakup Mechanisms in a Turbulent Flow (p. 794) (adapted from [366] (p. 1074)).

Figure 15.4: Bubble Breakup Mechanisms in a Turbulent Flow



These breakup mechanism stresses are expressed as:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

- Inertial force caused by turbulent velocity fluctuation:

$$\tau_{turb} = B_{turb}\rho_l(\varepsilon_l d_i)^{2/3} \quad for\ (d_i > \eta) \tag{15.32}$$

  where ρ_l is the liquid density, ε_l is the turbulent dissipation rate in the liquid phase, and η is the Kolmogorov microscale.

- Viscous shear force caused by velocity gradients in the bulk flow:

$$\tau_{shear} = B_{shear}\mu_l\dot{\gamma}_b \tag{15.33}$$

  where μ_l is the dynamic viscosity of the liquid and $\dot{\gamma}_b$ is the shear strain rate at the bulk.

- Viscous shear force caused by velocity gradients in eddies:

$$\tau_{eddy} = B_{eddy}\mu_l\dot{\gamma}_{eddy} \quad for\ (d_i \leq \eta) \tag{15.34}$$

  where $\dot{\gamma}_{eddy}$ is the characteristic strain rate of the flow in the smallest eddies.

- Interfacial drag (or friction):

$$\tau_{fric} = B_{fric}0.5\rho_l u_{T,i}^2 C_{D,i} \tag{15.35}$$

  where $u_{T,i}$ and $C_{D,i}$ are the terminal velocity and the drag coefficient of bubble i, respectively.

Constants B_{turb}, B_{shear}, B_{eddy}, and B_{fric} allow to consider uncertainties in the estimation of destroying stresses caused by different mechanisms. The default (and recommended by Liao [366] (p. 1074)) values for the coefficients are: B_{turb}=B_{shear}=B_{eddy}=1, and B_{fric} = 0.25.

15.2.2.2.6. Parabolic PDF

The breakage PDF function contains information on the probability of fragments formed by a breakage event. It provides the number of particles and the possible size distribution from the breakage. The parabolic form of the PDF implemented in Ansys Fluent enables you to define the breakage PDF such that

$$\beta(V|V') = 0.5\left[\frac{C}{V'} + \frac{1-C/2}{V'}\left\{24\left(\frac{V}{V'}\right)^2 - 24\left(\frac{V}{V'}\right) + 6\right\}\right] \tag{15.36}$$

where V and V' are the daughter and parent particle volumes, respectively. Depending on the value of the shape factor C, different behaviors will be observed in the shape of the particle breakage distribution function. For example, if C=2, the particle breakage has a uniform distribution. If $0<C<2$, a concave parabola is obtained, meaning that it is more likely to obtain unequally-sized fragments than equally-sized fragments. The opposite of this is true for $2<C<3$. Values outside of the range of 0 to 3 are not allowed, because the PDF cannot have a negative value.

Note that the PDF defined in Equation 15.36 (p. 795) is symmetric about $V/V'=0.5$.

15.2.2.2.7. Generalized PDF

The generalized form of the PDF implemented in Ansys Fluent enables you to simulate multiple breakage fragments (>2) and to specify the form of the daughter distribution (for example, uni-

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

form, equal-size, attrition, power law, parabolic, binary beta). The model itself can be applied to both the discrete method and the QMOM.

Considering the self-similar formulation [536] (p. 1084) where the similarity z is the ratio of daughter-to-parent size (that is, $z \equiv \frac{V}{V'}$), then the generalized PDF is given by

$$p\beta\left(V \middle| V'\right) = \frac{\theta(z)}{V'} \tag{15.37}$$

where $\theta(z)$ is the self-similar daughter distribution [141] (p. 1061). The k^{th} moment of $\theta(z)$, b_k, is

$$b_k = \int_0^1 z^k \theta(z) dz = \frac{B_k(V')}{V'^k} \tag{15.38}$$

where

$$B_k(V') = \int_0^{V'} V^k p\beta\left(V \middle| V'\right) dV \tag{15.39}$$

The conditions of number and mass conservation can then be expressed as

$$b_0 = \int_0^1 \theta(z) dz = p \tag{15.40}$$

$$b_1 = \int_0^1 z\theta(z) dz = 1 \tag{15.41}$$

The generalized form of $\theta(z)$ [141] (p. 1061) can be expressed as

$$\theta(z) = \sum_i w_i p_i \frac{z^{q_i - 1}(1-z)^{r_i - 1}}{\beta(q_i, r_i)} \tag{15.42}$$

where i can be 0 or 1, which represents $\theta(z)$ as consisting of 1 or 2 terms, respectively. For each term, w_i is the weighting factor, p_i is the averaged number of daughter particles, q_i and r_i are the exponents, and $\beta(q_i, r_i)$ is the beta function. The following constraints are imposed on the parameters in Equation 15.42 (p. 796) :

$$\sum_i w_i = 1 \tag{15.43}$$

$$\sum_i w_i p_i = p \tag{15.44}$$

$$\sum_i w_i \left(\frac{p_i q_i}{q_i + r_i}\right) = 1 \tag{15.45}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

In order to demonstrate how to transform the generalized PDF to represent an appropriate daughter distribution, consider the expressions shown in the tables that follow:

Table 15.3: Daughter Distributions

Type	$\theta(z)$
Equal-size [319] (p. 1071)	$p\delta\left(z-\frac{1}{p}\right)$
Attrition [319] (p. 1071)	$\delta(z-(1-\varepsilon))+\delta(z-\varepsilon)$
Power Law [671] (p. 1092)	$(\nu+1)\ z^{\nu-1}$
Parabolic -a [671] (p. 1092)	$(\nu+2)(\nu+1)z^{\nu-1}(1-z)$
Austin [34] (p. 1055)	$w(\nu_1+1)\ z^{\nu_1-1}+(1-w)(\nu_2+1)\ z^{\nu_2-1}$
Binary Beta -a [258] (p. 1068)	$60z^2(1-z)^2$
Binary Beta -b [417] (p. 1077)	$\frac{2}{\beta(\nu,\nu)}z^{\nu-1}(1-z)^{\nu-1}$
Uniform [671] (p. 1092)	$p(p-1)(1-z)^{p-2}$

Table 15.4: Daughter Distributions (cont.)

Type	p	Constraints
Equal-size [319] (p. 1071)	p	$p\geq 2$
Attrition [319] (p. 1071)	2	$\varepsilon \ll 1$
Power Law [671] (p. 1092)	$\frac{\nu+1}{\nu}$	$0<\nu\leq 1$
Parabolic -a [671] (p. 1092)	$\frac{\nu+2}{\nu}$	$0<\nu\leq 2$
Austin [34] (p. 1055)	$w\left(1+\frac{1}{\nu_1}\right)+(1-w)\left(1+\frac{1}{\nu_2}\right)$	$\nu_1,\nu_2>0;\ 1\geq w\geq \nu_1\left(\frac{\nu_2-1}{\nu_2-\nu_1}\right)$
Binary Beta -a [258] (p. 1068)	2	N/A
Binary Beta -b [417] (p. 1077)	2	ν 0
Uniform [671] (p. 1092)	p	$p\geq 2$

In Table 15.3: Daughter Distributions (p. 797), δ is the Dirac delta function, w is a weighting coefficient, and ε, ν, ν_1, and ν_2 are user-defined parameters.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

The generalized form can represent the daughter distributions in Table 15.3: Daughter Distributions (p. 797) by using the values shown in Table 15.5: Values for Daughter Distributions in General Form (p. 798).

Table 15.5: Values for Daughter Distributions in General Form

Type	$w0$	$p0$	$q0$	$r0$	$w1$	$p1$	$q1$	$r1$	Constraints
Equal-size	1	p	$\infty*$	$\infty*$	N/A	N/A	N/A	N/A	$p \geq 2$
Attrition	0.5	2	ε	1	0.5	2	1	ε	$\varepsilon \ll 1$
Power Law	1	$\frac{\nu+1}{\nu}$	ν	1	N/A	N/A	N/A	N/A	$0<\nu\leq 1$
Parabolic	1	$\frac{\nu+2}{\nu}$	ν	2	N/A	N/A	N/A	N/A	$0<\nu\leq 2$
Austin	w	$\frac{\nu_1+1}{\nu_1}$	ν_1	1	$1-w$	$\frac{\nu_2+1}{\nu_2}$	ν_2	1	$\nu_1,\nu_2>0$; $1\geq w\geq \nu_1\left(\frac{\nu_2-1}{\nu_2-\nu_1}\right)$
Binary Beta **	1	2	ν	ν	N/A	N/A	N/A	N/A	$\nu>0$
Uniform	1	p	1	p-1	N/A	N/A	N/A	N/A	$p \geq 2$

(*)You can approximate ∞ by using a very large number, such as 1e20.

(**)Binary Beta -a is a special case of Binary Beta -b when $\nu=3$.

Important:

Note that for the Ansys Fluent implementation of the generalized form of the PDF, you will only enter values for w_0, p_0, q_0, r_0, and q_1, and the remaining values (w_1, p_1, and r_1) will be calculated automatically.

15.2.2.2.8. Martinez-Bazan Breakage Kernel

The Martinez-Bazan breakage frequency [413] (p. 1077) is commonly used to model bubble breakage in stirred tank reactors. The kernel provides a phenomenological expression for the breakage rate of bubbles, which is based on the local values of the turbulence stress and surface tension. The breakage rate is expressed as:

$$g(V') = \begin{cases} K_g \frac{\sqrt{\beta(\varepsilon d_i)^{2/3} - 12\sigma/(\rho_l d_i)}}{d_i} & \text{for } d_i > d_{max} \\ 0 & \text{for } d_i \leq d_{max} \end{cases} \tag{15.46}$$

The maximum stable bubble diameter d_{max} is given by:

$$d_{max} = 2^{1/5} \frac{\sigma^{3/5}}{\rho_l^{3/5} \varepsilon^{2/5}} \tag{15.47}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

In the above equations,

K_g = 0.25 and β = 8.2 = constants found experimentally

d_i = parent diameter

ε = turbulent dissipation

σ = surface tension

ρ_l = liquid density

15.2.2.2.9. Inverted U-PDF

Most daughter distribution PDFs for bubble breakage assume a symmetric U-shaped distribution with the minimum probability of breakage occurring for equi-sized daughter bubbles. Although this kind of a daughter distribution function works well for many applications, an alternative inverted U-shaped distribution has also been proposed in [413] (p. 1077), where, by contrast with the U-shaped distribution, the probability of formation of equi-sized daughter bubbles is assumed to be at a maximum. In Ansys Fluent, this is implemented as:

$$\beta\left(V,\ V'\right)=\frac{\left(X^{2/3}-\Lambda^{5/3}\right)\left[\ \left(1-X^3\right)^{2/9}-\Lambda^{5/3}\right]}{\int_0^{V'}\left(X^{2/3}-\Lambda^{5/3}\right)\left[\ \left(1-X^3\right)^{2/9}-\Lambda^{5/3}\right]dV} \tag{15.48}$$

where $\Lambda=D_c/D_0$, $D_c=\left[\,12\sigma/\left(\beta\rho\right)\,\right]^{3/5}\varepsilon^{-2/5}$, $X=\left(\frac{V}{V'}\right)^{1/3}$, and $\beta\left(V,V'\right)$ is defined in Breakage (p. 790).

15.2.2.2.10. Aggregation

The aggregation kernel [397] (p. 1076) is expressed as

$$\Omega_{ag}=F_A a\left(V,V'\right) \tag{15.49}$$

where F_A is an aggregation factor for the calibration of the aggregation kernels.

The aggregation kernel has units of m^3/s, and is sometimes defined as a product of two quantities:

- the frequency of collisions between particles of volume V and particles of volume V'
- the "efficiency of aggregation" (that is, the probability of particles of volume V coalescing with particles of volume V').

The birth rate of particles of volume V due to aggregation is given by

$$B_{ag}=\frac{1}{2}\int_0^V a\left(V-V',V'\right)n\left(V-V'\right)n\left(V'\right)dV' \tag{15.50}$$

where particles of volume $V-V'$ aggregate with particles of volume V' to form particles of volume V. The factor $1/\ 2$ is included to avoid accounting for each collision event twice.

The death rate of particles of volume V due to aggregation is given by

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$D_{ag}=\int_0^{\infty} a\left(V,V'\right)n\left(V\right)n\left(V'\right)dV' \tag{15.51}$$

Important:

The breakage and aggregation kernels depend on the nature of the physical application. For example, in gas-liquid dispersion, the kernels are functions of the local liquid-phase turbulent dissipation.

The following is a list of aggregation functions available in Ansys Fluent:

- Constant
- Luo model
- Free molecular model
- Turbulent model
- Prince and Blanch model
- Liao model
- User-defined model

The Luo, free molecular, turbulent, Prince and Blanch, and Liao aggregation functions are described in detail in the sections that follow.

15.2.2.2.11. Luo Aggregation Kernel

For the Luo model [395] (p. 1076), the general aggregation kernel is defined as the rate of particle volume formation as a result of binary collisions of particles with volumes V_i and V_j:

$$a\left(V_i,V_j\right)=\omega_{ag}\left(V_i,V_j\right)P_{ag}\left(V_i,V_j\right)\left[m^3/\sec\right] \tag{15.52}$$

where $\omega_{ag}\left(V_i,V_j\right)\left[m^3/\ sec\right]$ is the frequency of collision and $P_{ag}\left(V_i,V_j\right)$ is the probability that the collision results in coalescence. The frequency is defined as follows:

$$\omega_{ag}\left(V_i,V_j\right)=\frac{\pi}{4}\left(d_i+d_j\right)^2\overline{u}_{ij} \tag{15.53}$$

where $\overline{u}_{ij}$ is the characteristic velocity of collision of two particles with diameters d_i and d_j.

$$\overline{u}_{ij}=\left(\overline{u}_i^2+\overline{u}_j^2\right)^{1/2} \tag{15.54}$$

where

$$\overline{u}_i=1.43\left(\varepsilon d_i\right)^{1/3} \tag{15.55}$$

The expression for the probability of aggregation is

$$P_{ag}=exp\left\{-c_1\frac{\left[0.75\left(1+x_{ij}^2\right)\left(1+x_{ij}^3\right)\right]^{1/2}}{\left(\rho_2/\rho_1+0.5\right)^{1/2}\left(1+x_{ij}\right)^3}We_{ij}^{1/2}\right\} \tag{15.56}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

where c_1 is a constant of order unity, $x_{ij}=d_i/\ d_j$, ρ_1 and ρ_2 are the densities of the primary and secondary phases, respectively, and the Weber number is defined as

$$We_{ij}=\frac{\rho_l d_i\left(\overline{u}_{ij}\right)^2}{\sigma} \tag{15.57}$$

15.2.2.2.12. Free Molecular Aggregation Kernel

Real particles aggregate and break with frequencies (or kernels) characterized by complex dependencies over particle internal coordinates [581] (p. 1086). In particular, very small particles (say up to $1\mu m$) aggregate because of collisions due to Brownian motions. In this case, the frequency of collision is size-dependent and usually the following kernel is implemented:

$$a\left(V_i,V_j\right)=\frac{2k_BT}{3\mu}\frac{\left(L_i+L_j\right)^2}{L_iL_j} \tag{15.58}$$

where k_B is the Boltzmann constant, T is the absolute temperature, μ is the viscosity of the suspending fluid, L_i and L_j are the diameters (or lengths) of particles i and j. This kernel is also known as the Brownian kernel or the perikinetic kernel.

15.2.2.2.13. Turbulent Aggregation Kernel

During mixing processes, mechanical energy is supplied to the fluid. This energy creates turbulence within the fluid. The turbulence creates eddies, which in turn help dissipate the energy. The energy is transferred from the largest eddies to the smallest eddies in which it is dissipated through viscous interactions. The size of the smallest eddies is the Kolmogorov microscale, η, which is expressed as a function of the kinematic viscosity and the turbulent energy dissipation rate:

$$\eta=\left(\frac{\nu^3}{\varepsilon}\right)^{1/4} \tag{15.59}$$

In the turbulent flow field, aggregation can occur by two mechanisms:

- viscous subrange mechanism: this is applied when particles are smaller than the Kolmogorov microscale, η
- inertial subrange mechanism: this is applied when particles are bigger than the Kolmogorov microscale. In this case, particles assume independent velocities.

For the viscous subrange, particle collisions are influenced by the local shear within the eddy. Based on work by Saffman and Turner [561] (p. 1085), the collision rate is expressed as,

$$a\left(V_i,V_j\right)=\varsigma_T\sqrt{\frac{8\pi}{15}}\dot{\gamma}\frac{\left(L_i+L_j\right)^3}{8} \tag{15.60}$$

where ς_T is a pre-factor that takes into account the capture efficiency coefficient of turbulent collision, and $\dot{\gamma}$ is the shear rate:

$$\dot{\gamma}=\frac{\varepsilon}{\nu}^{0.5} \tag{15.61}$$

For the inertial subrange, particles are bigger than the smallest eddy, therefore they are dragged by velocity fluctuations in the flow field. In this case, the aggregation rate is expressed using Abrahamson's model [6] (p. 1053),

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$a\left(V_i,V_j\right)=\varsigma_T 2^{3/2}\sqrt{\pi}\frac{\left(L_i+L_j\right)^2}{4}\sqrt{\left(U_i^2+U_j^2\right)} \tag{15.62}$$

where U_i^2 is the mean squared velocity for particle i.

The empirical capture efficiency coefficient of turbulent collision describes the hydrodynamic and attractive interaction between colliding particles. Higashitani et al. [244] (p. 1067) proposed the following relationship:

$$\varsigma_T=0.732\left(\frac{5}{N_T}\right)^{0.242};N_T\geq 5 \tag{15.63}$$

where N_T is the ratio between the viscous force and the Van der Waals force,

$$N_T=\frac{6\pi\mu\left(L_i+L_j\right)^3\dot{\lambda}}{8H} \tag{15.64}$$

Where H is the Hamaker constant, a function of the particle material, and $\dot{\lambda}$ is the deformation rate,

$$\dot{\lambda}=\left(\frac{4\varepsilon}{15\pi\nu}\right)^{0.5} \tag{15.65}$$

15.2.2.2.14. Prince and Blanch Aggregation Kernel

The aggregation model of Prince and Blanch [532] (p. 1084) assumes that the coalescence of two bubbles occurs in three steps:

1. Bubbles collide trapping a small amount of liquid between them.
2. The liquid film separating the bubbles drains until it reaches a critical thickness.
3. The film ruptures, and the bubbles join into one large bubble.

This aggregation kernel is modeled by a collision rate of two bubbles and a collision efficiency related to the time required for coalescence:

$$\mathrm{a}\left(V_i,V_j\right)=\left(\theta_{ij}^T+\theta_{ij}^B+\theta_{ij}^S\right)\eta_{ij} \tag{15.66}$$

The collision efficiency is modeled by comparing the time required for coalescence t_{ij} and the actual contact time during the collision τ_{ij}:

$$\eta_{ij}=e^{-t_{ij}/\tau_{ij}} \tag{15.67}$$

where:

$$t_{ij}=\left(\frac{\rho_l\, r_{ij}^3}{16\ \sigma}\right)^{1/2}\ln\left(\frac{h_o}{h_f}\right) \tag{15.68}$$

and:

$$\tau_{ij}=\frac{r_{ij}^{2/3}}{\varepsilon_l^{1/3}} \tag{15.69}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

In Equation 15.68 (p. 802) and Equation 15.69 (p. 802), ρ_l is the density of the liquid, σ is the surface tension, h_o is the initial film thickness, h_f is the critical film thickness when rupture occurs, ε_l is the turbulent eddy dissipation of the liquid, and r_{ij} is the equivalent radius defined as:

$$r_{ij}=\left(\frac{1}{2}\left(\frac{1}{r_i}+\frac{1}{r_j}\right)\right)^{-1} \tag{15.70}$$

where r_i and r_j are the radii of bubbles i and j, respectively.

The turbulent contributions to collision frequency are modeled as:

$$\theta_{ij}^{T}=F_{CT}S_{ij}\left(u_{ti}^2+u_{tj}^2\right)^{1/2} \tag{15.71}$$

where the cross-sectional area of the colliding particles is defined by:

$$S_{ij}=\frac{\pi}{4}\left(d_i+d_j\right)^2 \tag{15.72}$$

the turbulent velocity is given by:

$$u_{ti}=\sqrt{2}\,\varepsilon_l^{1/3}d_i^{1/3} \tag{15.73}$$

and F_{CT} is a calibration factor for the turbulent contribution.

The buoyancy contribution to the collision frequency is calculated as:

$$\theta_{ij}^{B}=F_{CB}S_{ij}\left|U_{rj}-U_{ri}\right| \tag{15.74}$$

where:

$$U_{ri}=\sqrt{\frac{2.14\ \sigma}{\rho_l\ d_i}+0.505gd_i} \tag{15.75}$$

and F_{CB} is a calibration factor for the buoyancy.

In Equation 15.72 (p. 803), Equation 15.73 (p. 803), and Equation 15.75 (p. 803), d_i and d_j are the diameters of bubbles i and j, respectively, and g is gravity.

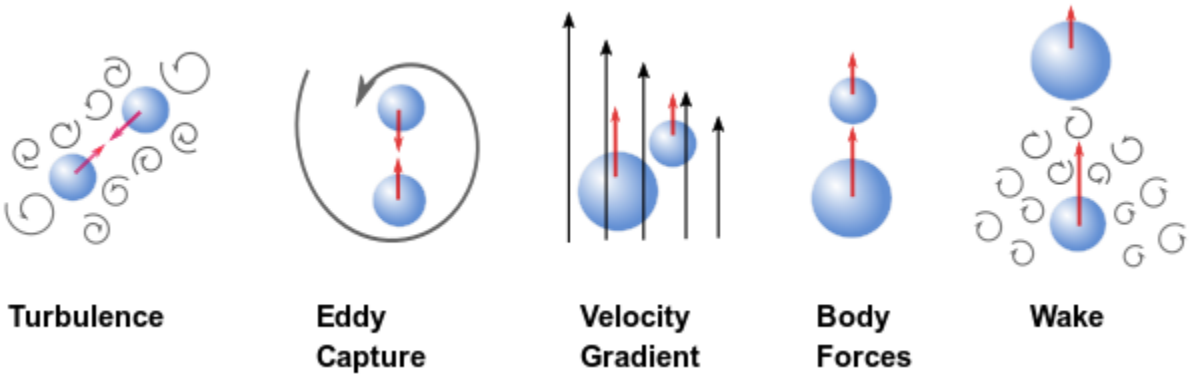
The shear contribution to collision frequency (θ_{ij}^{S} in Equation 15.66 (p. 802)) is currently neglected.

15.2.2.2.15. Liao Aggregation Kernel

Coalescence occurs where two or more fluid particles collides. This can happen for various reasons such as turbulence, eddy capture, velocity gradient in the mean flow, body forces, and wake entrainment [366] (p. 1074) as shown in Figure 15.5: Bubble Collision in a Turbulent Flow (p. 804) (adapted from [366] (p. 1074)).

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Figure 15.5: Bubble Collision in a Turbulent Flow



Assuming the cumulative contribution of all the coalescence mechanisms, Liao [366] (p. 1074) proposed a formulation for the total coalescence frequency a of a binary collision between two bubbles of volumes V_i and V_j:

$$\mathrm{a}\left(V_i, V_j\right)=\frac{\alpha_{g,max}}{\alpha_{g,max}+\alpha_g}\left(\Gamma_{turb}+\Gamma_{eddy}+\Gamma_{shear}+\Gamma_{buoy}+\Gamma_{wake}\right) \tag{15.76}$$

where α_g is the bubble packing limit, and $\alpha_{g,max}$ is its maximum. When the local volume fraction of the bubble phase approaches the maximum packing limit $\alpha_{g,max}$, the coalescence rate increases infinitely.

Since bubble-bubble collision can result either in coalescence or bouncing of the colliding bubbles, the coalescence efficiency (or probability) is used. Coalescence frequency a is represented as the product of the collision frequency h and the efficiency of each collision event λ:

$$a=h\lambda \tag{15.77}$$

Similar to the kinetic gas theory, the collision frequency h is approximated as the volume swept by the two particles in unit time:

$$h=Au_{rel} \tag{15.78}$$

where u_{rel} is the relative velocity between the two bubbles, and A is the effective cross-sectional area of collision.

For a random bubble collision caused by turbulent fluctuation, the effective cross-sectional area is:

$$A_{turb}=\frac{\pi}{4}\left(d_i+d_j\right)^2 \tag{15.79}$$

where d_i and d_j are diameters of bubbles i and j, respectively.

The relative velocity between bubbles i and j is estimated by:

$$u_{rel,turb}=\left(u_{turb,i}^2+u_{turb,j}^2\right)^{1/2} \tag{15.80}$$

where u_{turb} is the turbulent velocity over a distance d_i given by:

$$u_{turb}=\sqrt{2}\left(\varepsilon_l d_i\right)^{1/3} \tag{15.81}$$

where ε_l is the turbulent dissipation rate in the liquid phase.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

For a turbulence-induced collision, the coalescence frequency Γ_{turb} between bubble i and bubble j is obtained as:

$$\Gamma_{turb}=\frac{\pi}{4}\left(d_i+d_j\right)^2 C_{turb}\sqrt{2}\left(d_i^{2/3}+d_j^{2/3}\right)^{1/2}\varepsilon_l^{1/3}\lambda_{inertial} \tag{15.82}$$

where C_{turb} is a constant, and $\lambda_{inertial}$ is the coalescence efficiency in the inertial collision regime. Equation 15.82 (p. 805) is valid for bubbles much larger than the Kolmogorov length scale.

When the buoyancy effects are considered, collision can occur only when the faster-moving bubble approaches the slower-moving bubble from bellow in an upward flow. The coalescence frequency Γ_{buoy} is modeled as:

$$\Gamma_{buoy}=0.5\frac{\pi}{4}\left(d_i+d_j\right)^2 C_{buoy}\left|u_{T,i}-u_{T,j}\right|\lambda_{eff} \tag{15.83}$$

where C_{buoy} is a constnat, and $u_{T,i}$ and $u_{T,j}$ are the terminal velocities of bubble i and bubble j, respectively. The effective coalescence efficiency λ_{eff} is calculated by:

$$\lambda_{eff}=\begin{cases}\lambda_{inertial} & for\ \left(d_i+d_j>\ \eta\right)\\ \lambda_{viscous} & for\ \left(d_i+d_j\leq\ \eta\right)\end{cases} \tag{15.84}$$

where η is the Kolmogorov length scale.

The coalescence efficiency for an inertial collision $\lambda_{inertial}$ is expressed as:

$$\lambda_{inertial}=exp\left(-C_{eff}We_{ij,max}^{0.5}\right) \tag{15.85}$$

where the coefficient C_{eff} has a default value of 5.0, and

$$We_{ij,max}=\frac{\rho_l d_{eq}}{\sigma}\ max\left(u_{rel,turb},u_{rel,buoy},u_{rel,shear}\right)^2 \tag{15.86}$$

where ρ_l is the liquid density, σ is the surface tension coefficient, and

$$d_{eq}=\frac{2d_i d_j}{d_i+d_j} \tag{15.87}$$

The coalescence efficiency of the viscous collision $\lambda_{viscous}$ is expressed as:

$$\lambda_{viscous}=exp\left(-\frac{3\mu_l d_{eq}\dot{\gamma}_{eddy}}{4\sigma}\ln\left(\frac{\pi\sigma d_{eq}^2}{32A_H}\right)^{1/3}\right) \tag{15.88}$$

where μ_l is the dynamic viscosity of the liquid, A_H is the Hamaker constant that represents the strength of the van der Waals interactions between macroscopic bodies, and $\dot{\gamma}_{eddy}$ is the characteristic strain rate of the flow in the smallest eddies. For an air-water interface, its value is approximated to 3.7 x 10^{-20} J.

In a shear-induced collision, a bubble can only collide with another bubble it follows in the direction of the bulk velocity from the higher-velocity side. The coalescence frequency Γ_{shear} between bubble i and bubble j caused by the velocity gradient is expressed as:

$$\Gamma_{shear}=0.5\frac{\pi}{4}\left(d_i+d_j\right)^2 C_{shear}\left[\frac{0.5}{\pi}\left(d_i+d_j\right)\dot{\gamma}_b\right]\lambda_{eff} \tag{15.89}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

where C_{shear} is a constant, $\dot{\gamma}_b$ is the shear strain rate of the bulk flow, and λ_{eff} is defined by Equation 15.84 (p. 805).

For bubbles that are much smaller than the Kolmogorov microscale, the collision is assumed to be controlled by viscous forces. The coalescence frequency Γ_{eddy} is modeled as:

$$\Gamma_{eddy}=0.5\frac{\pi}{4}\left(d_i+d_j\right)^2 C_{eddy}\left[\frac{0.5}{\pi}\left(d_i+d_j\right)\dot{\gamma}_{eddy}\right]\lambda_{viscous} \tag{15.90}$$

where C_{eddy} is a constant. It can be derived from the Kolmogorov scales to be:

$$\dot{\gamma}_{eddy}=\sqrt{\frac{\rho_l \varepsilon_l}{\mu_l}} \tag{15.91}$$

When a bubble of a size bigger than a critical diameter moves through a liquid, it generates a wake in the liquid phase behind it. The trailing bubbles within the wake region of the leading bubble can be accelerated and reach the leading bubble. The coalescence efficiency for wake-entrainment Γ_{wake} is modeled as:

$$\Gamma_{wake}=C_{wake}0.5\frac{\pi}{4}\left[d_i^2 u_{T,i} C_{D,i}^{1/3}\Theta_i+d_j^2 u_{T,j} C_{D,j}^{1/3}\Theta_j\right] \tag{15.92}$$

where C_{wake} is a constant, $u_{T,i}$ and $u_{T,j}$ are the terminal velocities of bubbles i and j, respectively, and $C_{D,i}$ and $C_{D,j}$ are the drag coefficients of bubbles i and j, respectively. The blending function Θ_i for bubble i is defined as:

$$\Theta_i=\begin{cases}\dfrac{\left(d_i-0.5d_{crit}\right)^6}{\left(d_i-0.5d_{crit}\right)^6+\left(0.5d_{crit}\right)^6} & for\ \ d_i\geq 0.5d_{crit}\\ 0 & for\ \ d_i<0.5d_{crit}\end{cases} \tag{15.93}$$

where the critical diameter d_{crit} is computed as:

$$d_{crit}=4\sqrt{\frac{\sigma}{g\left(\rho_l-\rho_g\right)}} \tag{15.94}$$

where ρ_g is the gas density.

The constants C_{turb}, C_{buoy}, C_{shear}, C_{eddy}, and C_{wake} account for uncertainty in the estimation of relative velocity under different conditions. Liao [366] (p. 1074) recommends setting these coefficients to 1.0, which are the default settings in Ansys Fluent.

15.2.2.3. Particle Birth by Nucleation

Depending on the application, spontaneous nucleation of particles can occur due to the transfer of molecules from the primary phase. For example, in crystallization from solution, the first step is the phase separation or "birth" of new crystals. In boiling applications, the creation of the first vapor bubbles is a nucleation process referred to as nucleate boiling.

The nucleation rate is defined through a boundary condition as shown in Equation 15.10 (p. 789).

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

15.2.3. Solution Methods

As discussed, the population balance equation can be solved by the four different methods in Ansys Fluent: the discrete method, the inhomogeneous discrete method, the standard method of moments (SMM), and the quadrature method of moments (QMOM). For each method, the Ansys Fluent implementation is limited to a single internal coordinate corresponding to particle size. The following subsections describe the theoretical background of each method and list their advantages and disadvantages.

15.2.3.1. The Discrete Method and the Inhomogeneous Discrete Method

15.2.3.2. The Standard Method of Moments (SMM)

15.2.3.3. The Quadrature Method of Moments (QMOM)

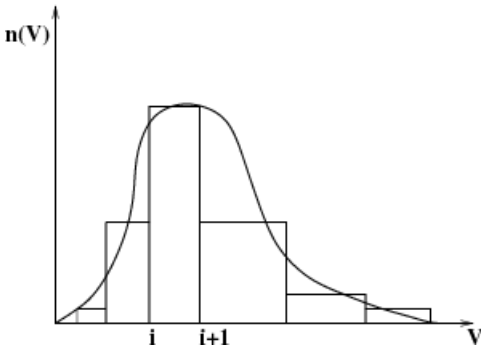
15.2.3.4. The Direct Quadrature Method of Moments (DQMOM)

15.2.3.1. The Discrete Method and the Inhomogeneous Discrete Method

The discrete method (also known as the classes or sectional method) was developed by Hounslow [255] (p. 1067), Litster [381] (p. 1075), and Ramkrishna [536] (p. 1084). It is based on representing the continuous particle size distribution (PSD) in terms of a set of discrete size classes or bins, as illustrated in Figure 15.6: A Particle Size Distribution as Represented by the Discrete Method (p. 807). The advantages of this method are its robust numerics and that it gives the PSD directly. The disadvantages are that the bins must be defined *a priori* and that a large number of classes may be required.

The solution methods for the inhomogeneous discrete method are based on the discrete method and therefore share many of the same fundamentals.

Figure 15.6: A Particle Size Distribution as Represented by the Discrete Method



15.2.3.1.1. Numerical Method

In Ansys Fluent, the PBE is written in terms of volume fraction of particle size i:

$$\frac{\partial}{\partial t}\left(\rho_s \alpha_i\right)+\nabla\cdot\left(\rho_s u_i \alpha_i\right)+\frac{\partial}{\partial V}\left(\frac{G_v \rho_s \alpha_i}{V}\right) = \rho_s V_i\left(B_{ag,i}-D_{ag,i}+B_{br,i}-D_{br,i}\right) + 0^i \rho_s V_0 \dot{n}_0 \tag{15.95}$$

where ρ_s is the density of the secondary phase and α_i is the volume fraction of particle size i, defined as

$$\alpha_i = N_i V_i \qquad i=0,1,\cdots,N-1 \tag{15.96}$$

where

$$N_i(t) = \int_{V_i}^{V_{i+1}} n(V,t)\,dV \tag{15.97}$$

and V_i is the volume of the particle size i. In Ansys Fluent, a fraction of α, called f_i, is introduced as the solution variable. This fraction is defined as

$$f_i = \frac{\alpha_i}{\alpha} \tag{15.98}$$

where α is the total volume fraction of the secondary phase.

The nucleation rate $\dot{n}_0$ appears in the discretized equation for the volume fraction of the smallest size V_0. The notation 0^i signifies that this particular term, in this case $\rho_s V_0 \dot{n}_0$, appears in Equation 15.95 (p. 807) only in the case of the smallest particle size.

The growth rate in, Equation 15.95 (p. 807) [255] (p. 1067), is discretized as follows:

$$\frac{\partial}{\partial V}\left(\frac{G_v \rho_s \alpha_i}{V}\right) = \rho_s V_i \left[\left(\frac{G_{v,i-1} N_{i-1}}{V_i - V_{i-1}}\right) - \left(\frac{G_{v,i} N_i}{V_{i+1} - V_i}\right)\right] \tag{15.99}$$

The volume coordinate is discretized as [255] (p. 1067) $V_{i+1} / V_i = 2^q$ where $q = 1,2,...$ and is referred to as the "ratio factor".

The particle birth and death rates are defined as follows:

$$B_{ag,i} = \sum_{k=1}^{N}\sum_{j=1}^{N} a_{kj} N_k N_j x_{kj} \xi_{kj} \tag{15.100}$$

$$D_{ag,i} = \sum_{j=1}^{N} a_{ij} N_i N_j \tag{15.101}$$

$$B_{br,i} = \sum_{j=i+1}^{N} g(V_j) N_j \beta(V_i | V_j) \tag{15.102}$$

$$D_{br,i} = g(V_i) N_i \tag{15.103}$$

where $a_{ij} = a(V_i, V_j)$ and

$$\xi_{kj} = \begin{cases} 1 & \text{for } V_i < V_{ag} < V_{i+1}, \text{ where } i \le N-1 \\ 0 & \text{otherwise} \end{cases} \tag{15.104}$$

V_{ag} is the particle volume resulting from the aggregation of particles k and j, and is defined as

$$V_{ag} = \left[x_{kj} V_i + (1 - x_{kj}) V_{i+1}\right] \tag{15.105}$$

where

$$x_{kj} = \frac{V_{ag} - V_{i+1}}{V_i - V_{i+1}} \tag{15.106}$$

If V_{ag} is greater than or equal to the largest particle size V_N, then the contribution to class $N-1$ is

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$x_{kj}=\frac{V_{ag}}{V_N} \tag{15.107}$$

Important:

Note that there is no breakage for the smallest particle class.

15.2.3.1.2. Breakage Formulations for the Discrete Method

The default breakage formulation for the discrete method in Ansys Fluent is based on the Hagesather method [223] (p. 1065). In this method, the breakage sources are distributed to the respective size bins, preserving mass and number density. For the case when the ratio between successive bin sizes can be expressed as 2^n where $n=1,2,...$, the source in bin i, ($i=1,\ldots,N$) can be expressed as

$$\begin{aligned} B_b(i) &= \sum_N^{k=i+1,i\neq N}\Omega_b(v_k,v_i)+\sum_i^{k=i,i\neq N}x_{i+1,k}\Omega_b(v_{i+1},v_k) \\ &+\sum_{i-1}^{k=1,i\neq 1}(1-x_{i,k})\Omega_b(v_i,v_k) \end{aligned} \tag{15.108}$$

Here

$$\Omega_b(v_k,v_i)=N_k g(v_k)\beta(v_k,v_i) \tag{15.109}$$

A more mathematically rigorous formulation is given by Ramkrishna [324] (p. 1072), where the breakage rate is expressed as

$$B_b(i)=\sum_i^N n_{i,k}g(v_k)N_k \tag{15.110}$$

where

$$n_{i,k}=\int_{v_i} v_{i+1}\frac{v_{i+1}-v}{v_{i+1}-v_i}\beta(v_k,v)dv+\int_{v_{i-1}}^{v_i}\frac{v-v_{i-1}}{v_i-v_{i-1}}\beta(v_k,v)\,dv \tag{15.111}$$

The Ramkrishna formulation can be slow due to the large number of integration points required. However, for simple forms of β, the integrations can be performed relatively easily. The Hagesather formulation requires fewer integration points and the difference in accuracy with the Ramkrishna formulation can be corrected by a suitable choice of bin sizes.

Important:

To keep the computing time reasonable, a volume averaged value is used for the turbulent eddy dissipation when the Luo model is used in conjunction with the Ramkrishna formulation.

Note:

The inhomogeneous discrete phase applies the Hagesather formulation.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

15.2.3.2. The Standard Method of Moments (SMM)

The SMM, proposed by Randolph and Larson [537] (p. 1084) is an alternative method for solving the PBE. Its advantages are that it reduces the dimensionality of the problem and that it is relatively simple to solve transport equations for lower-order moments. The disadvantages are that exact closure of the right-hand side is possible only in cases of constant aggregation and size-independent growth, and that breakage modeling is not possible. The closure constraint is overcome, however, through QMOM (see The Quadrature Method of Moments (QMOM) (p. 811)).

15.2.3.2.1. Numerical Method

The SMM approach is based on taking moments of the PBE with respect to the internal coordinate (in this case, the particle size L).

Defining the k^{th} moment as

$$m_k(\vec{x},t)=\int_0^\infty n(L;\vec{x},t)L^k dL \qquad k=0,1,\cdots,N-1 \tag{15.112}$$

and assuming constant particle growth, its transport equation can be written as

$$\frac{\partial}{\partial t}(\rho_s m_k)+\nabla\cdot(\rho_s\vec{u}m_k)=\rho_s(\overline{B}_{\text{ag},k}-\overline{D}_{\text{ag},k}+\overline{B}_{\text{br},k}-\overline{D}_{\text{br},k})+\rho_s\left(0^k\dot{n}_0+\text{Growth}\right) \tag{15.113}$$

where

$$\overline{B}_{ag,k} = \tfrac{1}{2}\int_0^\infty n(\lambda)\int_0^\infty a(u,\lambda)(u,\lambda)\left(u^3+\lambda^3\right)^{k/3}n(u)du d\lambda \tag{15.114}$$

$$\overline{D}_{ag,k} = \int_0^\infty L^k n(L)\int_0^\infty a(L,\lambda)n(\lambda)d\lambda dL \tag{15.115}$$

$$\overline{B}_{br,k} = \int_0^\infty L^k\int_0^\infty g(\lambda)\beta(L|\lambda)n(\lambda)d\lambda dL \tag{15.116}$$

$$\overline{D}_{br,k} = \int_0^k L^k g(L)n(L)d \tag{15.117}$$

N is the specified number of moments and $\dot{n}_0$ is the nucleation rate. The growth term is defined as

$$\text{Growth}\equiv\int_0^\infty kL^{k-1}G(L)n(L,t)dL \tag{15.118}$$

and for constant growth is represented as

$$kGm_{k-1} \tag{15.119}$$

Equation 15.114 (p. 810) can be derived by using

$$u^3=L^3-\lambda^3;dL=\frac{u^2}{L^2}du$$

and reversing the order of integration. From these moments, the parameters describing the gross properties of particle population can be derived as

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$N_{total} = m_0 \tag{15.120}$$

$$L_{total} = m_1 \tag{15.121}$$

$$A_{total} = K_a m_2 \tag{15.122}$$

$$V_{total} = K_v m_3 \tag{15.123}$$

$$d_{32} = \frac{m_3}{m_2} \tag{15.124}$$

These properties are related to the total number, length, area, and volume of solid particles per unit volume of mixture suspension. The Sauter mean diameter, d_{32}, is usually used as the mean particle size.

To close Equation 15.113 (p. 810), the quantities represented in Equation 15.114 (p. 810) – Equation 15.117 (p. 810) need to be expressed in terms of the moments being solved. To do this, one approach is to assume size-independent kernels for breakage and aggregation, in addition to other simplifications such as the Taylor series expansion of the term $\left(u^3+\lambda^3\right)^{k/3}$. Alternatively, a profile of the PSD could be assumed so that Equation 15.114 (p. 810) – Equation 15.117 (p. 810) can be integrated and expressed in terms of the moments being solved.

In Ansys Fluent, an exact closure is implemented by restricting the application of the SMM to cases with size-independent growth and a constant aggregation kernel.

15.2.3.3. The Quadrature Method of Moments (QMOM)

The quadrature method of moments (QMOM) was first proposed by McGraw [419] (p. 1077) for modeling aerosol evolution and coagulation problems. Its applications by Marchisio et al. [410] (p. 1076) have shown that the method requires a relatively small number of scalar equations to track the moments of population with small errors.

The QMOM provides an attractive alternative to the discrete method when aggregation quantities, rather than an exact PSD, are desired. Its advantages are fewer variables (typically only six or eight moments) and a dynamic calculation of the size bins. The disadvantages are that the number of abscissas may not be adequate to describe the PSD and that solving the Product-Difference algorithm may be time consuming.

15.2.3.3.1. Numerical Method

The quadrature approximation is based on determining a sequence of polynomials orthogonal to $n(L)$ (that is, the particle size distribution). If the abscissas of the quadrature approximation are the nodes of the polynomial of order N, then the quadrature approximation

$$\int_0^\infty f(L)n(L)dL \approx \sum_{i=1}^{N} f(L_i)w_i, \tag{15.125}$$

is exact if $f(L)$ is a polynomial of order N or smaller [140] (p. 1061). In all other cases, the closer $f(L)$ is to a polynomial, the more accurate the approximation.

A direct way to calculate the quadrature approximation is by means of its definition through the moments:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$m_k = \sum_{i=1}^{N} w_i L_i^k. \tag{15.126}$$

The quadrature approximation of order N is defined by its N weights w_i and N abscissas L_i and can be calculated by its first $2N$ moments $m_0, \ldots, m_{2N-1}$ by writing the recursive relationship for the polynomials in terms of the moments m_k. Once this relationship is written in matrix form, it is easy to show that the roots of the polynomials correspond to the eigenvalues of the Jacobi matrix [531] (p. 1084). This procedure is known as the moment inversion algorithm, which uses either the Product-Difference algorithm [213] (p. 1065) or the Wheeler algorithm [702] (p. 1093). Once the weights and abscissas are known, the source terms due to coalescence and breakage can be calculated and therefore the transport equations for the moments can be solved.

Applying Equation 15.125 (p. 811) and Equation 15.126 (p. 812), the birth and death terms in Equation 15.113 (p. 810) can be rewritten as

$$\overline{B}_{ag,k} = \frac{1}{2} \sum_{i=1}^{N} w_i \sum_{j=1}^{N} w_j \left(L_i^3 + L_j^3 \right)^{k/3} a\left(L_i, L_j \right) \tag{15.127}$$

$$\overline{D}_{ag,k} = \sum_{i=1}^{N} L_i^k w_i \sum_{j=1}^{N} w_j a\left(L_i, L_j \right) \tag{15.128}$$

$$\overline{B}_{br,k} = \sum_{i=1}^{N} w_i \int_0^{\infty} L_k g\left(L_i \right) \beta\left(L \middle| L_i \right) dL \tag{15.129}$$

$$\overline{D}_{br,k} = \sum_{i=1}^{N} w_i L_i^k g\left(L_i \right) \tag{15.130}$$

Theoretically, there is no limitation on the expression of breakage and aggregation kernels when using QMOM.

The nucleation rate is defined in the same way as for the SMM. The growth rate for QMOM is defined by Equation 15.118 (p. 810) and represented as

$$\sum_{i=1}^{N} w_i L_i^{k-1} G\left(L_i \right) \tag{15.131}$$

to allow for a size-dependent growth rate.

15.2.3.4. The Direct Quadrature Method of Moments (DQMOM)

DQMOM equations are derived from the basic number density function equation via the moment transfer method, in a similar way to QMOM. The difference is that DQMOM assumes that each Quadrature point will occupy an independent velocity field, whereas QMOM assumes that all Quadrature points are moving on the same velocity field. This difference enables DQMOM to predict particle segregation due to particle interaction.

In this implementation of DQMOM, four phases must be specified: one primary phase and three secondary phases that are DQMOM phases. Compared to QMOM, for a three Quadrature points system, the DQMOM method only needs three extra equations to solve for the effective length of the particle, but there are additional source terms for the volume fraction equation for each DQMOM phase. In Ansys Fluent, three particle interactions are accounted for, growth, aggregation, and breakage. Nucleation is not considered.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

15.2.3.4.1. Numerical Method

The DQMOM equations, describing a poly-dispersed particle system undergoing aggregation, breakage, and growth can be written as follows (the details of the DQMOM formulation can be found in [169] (p. 1062)):

$$\frac{\partial \varepsilon_i \rho_s}{\partial t} + \nabla \cdot (\vec{u}_{si} \varepsilon_i \rho_s) = 3 k_v \rho_s L_i^2 (b_i + w_i G_i) - 2 k_v \rho_s L_i^3 a_i \tag{15.132}$$

$$\frac{\partial \varepsilon_i L_i \rho_s}{\partial t} + \nabla \cdot (\vec{u}_{si} \varepsilon_i L_i \rho_s) = 4 k_v \rho_s L_i^3 (b_i + w_i G_i) - 3 k_v \rho_s L_i^4 a_i \tag{15.133}$$

where ε_i and $\varepsilon_i L_i$ are the VOF and the effective length of the particle phase, respectively. w_i is the number of particles per unit volume and G_i is the growth rate at Quadrature point i, while a_i and b_i can be computed through a linear system resulting from the moment transformation of the particle number density transport equation using N Quadrature points. The linear system can be written in matrix form as

$$A\alpha = d \tag{15.134}$$

Where the $2N \times 2N$ coefficient matrix $A = \left[A_1 A_2\right]$ is defined by

$$A_1 = \begin{bmatrix} 1 & \dots & 1 \\ 0 & \dots & 0 \\ -L_1^2 & \dots & -L_N^2 \\ \vdots & \ddots & \vdots \\ 2(1-N)L_1^{2N-1} & \dots & 2(1-N)L_N^{2N-1} \end{bmatrix} \tag{15.135}$$

$$A_2 = \begin{bmatrix} 0 & \dots & 0 \\ 1 & \dots & 1 \\ 2L_1 & \dots & L_N \\ \vdots & \ddots & \vdots \\ (2N-1)L_1^{2N-2} & \dots & (2N-1)L_N^{2N-2} \end{bmatrix} \tag{15.136}$$

The $2\mathrm{N}$ vector of unknowns α is defined by

$$\alpha = [a_1 \ \dots \ a_N \ \ b_1 \ \dots \ b_N]^T = \begin{bmatrix} a \\ b \end{bmatrix} \tag{15.137}$$

The right hand side of Equation 15.134 (p. 813) is the known source terms involving aggregation and breakage phenomena only. The growth term is accounted for directly in Equation 15.132 (p. 813) and Equation 15.133 (p. 813). At present, nucleation is not considered.

$$d = \left[S_0^{(N)} \ \dots \ S_{2N-1}^{(N)}\right]^T \tag{15.138}$$

The source term for the k^{th} moment $S_k^{(N)}\left(k = 0, \dots, 2N-1\right)$ is defined as

$$S_k^{(N)}(x,t) = \int_0^\infty L^k S(x,t) dL \tag{15.139}$$

When the abscissas of the Quadrature points L_i are distinct, the matrix A is well defined and a unique solution of Equation 15.134 (p. 813) can be obtained. Otherwise, the matrix A is not full rank and cannot be inverted to find a unique solution for α. The method adopted by Ansys Fluent to overcome this problem is to employ a perturbation technique. For example, for the current three Quadrature points system, the perturbation technique will add a small value to the abscissas to make sure the matrix A is full rank. It is important to note that the perturbation technique is

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

only used for the definition of matrix A and no modifications are made to the source term vector of Equation 15.139 (p. 813). Therefore, both the weights and overall source terms resulting from aggregation and breakage are not affected by the perturbation method. The simulation tests have found that the perturbation method can stabilize the solutions of Equation 15.134 (p. 813) and reduce the physically unrealistically large source terms for the two phases whose abscissa are too close in value. However, the technique has little effect on the phase whose abscissa L_i is distinct from the other two.

15.2.4. Population Balance Statistics

The following sections introduce statistical concepts that are useful when using the population balance models.

15.2.4.1. Reconstructing the Particle Size Distribution from Moments

15.2.4.2. The Log-Normal Distribution

15.2.4.1. Reconstructing the Particle Size Distribution from Moments

Given a set of moments, the most likely PSD can be obtained based on the "statistically most probable" distribution for turbulent flames [526] (p. 1083), which was adapted for crystallization problems by Baldyga and Orciuch [39] (p. 1055).

The number density function $n(L)$ is expressed as

$$n(L) = \exp\left(\sum_{i=0}^{N-1} A_i L^i\right) \tag{15.140}$$

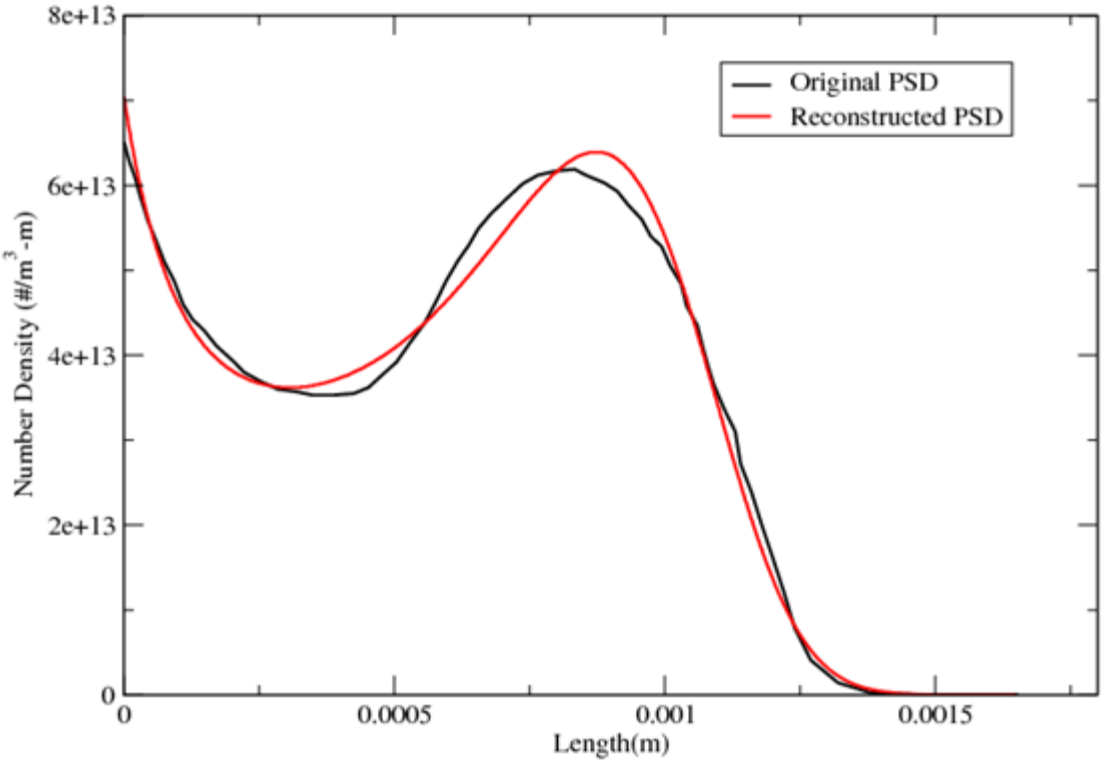
The equation for the k^{th} moment is now written as

$$m_k = \int_0^\infty L^k \exp\left(\sum_{i=0}^{N-1} A_i L^i\right) dL \qquad k = 0, 1, \cdots, N-1 \tag{15.141}$$

Given N moments, the coefficients A_i can be found by a globally convergent Newton-Raphson method to reconstruct the particle size distribution (for example, Figure 15.7: Reconstruction of a Particle Size Distribution (p. 815)).

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Figure 15.7: Reconstruction of a Particle Size Distribution



15.2.4.2. The Log-Normal Distribution

When using either of the discrete population balance methods, you have the option of specifying the size distribution at a velocity inlet by specifying a log-normal distribution.

The log-normal distribution for the number density, n, as a function of the particle size, L, can be written as:

$$n(L)=\frac{1}{L\sigma\sqrt{2\pi}}e^{\frac{(\ln L-\mu)^2}{2\sigma^2}} \tag{15.142}$$

where μ and σ are, respectively, the location and scale parameters of the distribution and can be written as

$$\mu=\ln(\mu')-\frac{1}{2}\ln\left(1+\frac{\sigma'^2}{\mu'^2}\right)$$

$$\sigma^2=\ln\left(1+\frac{\sigma'^2}{\mu'^2}\right)$$

μ' and σ' are the mean and standard deviation, respectively, and are specified in the boundary conditions as shown in Initializing Bin Fractions With a Log-Normal Distribution.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information
of ANSYS, Inc. and its subsidiaries and affiliates.

Chapter 16: Solidification and Melting

This chapter describes how you can model solidification and melting in Ansys Fluent. For information about using the model, see Modeling Solidification and Melting of the User's Guide. Information about the theory behind the model is organized into the following sections:

16.1. Overview

16.2. Limitations

16.3. Introduction

16.4. Energy Equation

16.5. Momentum Equations

16.6. Turbulence Equations

16.7. Species Equations

16.8. Back Diffusion

16.9. Pull Velocity for Continuous Casting

16.10. Contact Resistance at Walls

16.11. Thermal and Solutal Buoyancy

16.1. Overview

Ansys Fluent can be used to solve fluid flow problems involving solidification and/or melting taking place at one temperature (for example, in pure metals) or over a range of temperatures (for example, in binary alloys). Instead of tracking the liquid-solid front explicitly, Ansys Fluent uses an enthalpy-porosity formulation. The liquid-solid mushy zone is treated as a porous zone with porosity equal to the liquid fraction, and appropriate momentum sink terms are added to the momentum equations to account for the pressure drop caused by the presence of solid material. Sinks are also added to the turbulence equations to account for reduced porosity in the solid regions.

Ansys Fluent provides the following capabilities for modeling solidification and melting:

- calculation of liquid-solid solidification/melting in pure metals as well as in binary alloys
- modeling of continuous casting processes (that is, "pulling" of solid material out of the domain)
- modeling of the thermal contact resistance between solidified material and walls (for example, due to the presence of an air gap)
- modeling of species transport with solidification/melting
- postprocessing of quantities related to solidification/melting (that is, liquid fraction and pull velocities)

These modeling capabilities allow Ansys Fluent to simulate a wide range of solidification/melting problems, including melting, freezing, crystal growth, and continuous casting. The physical equations

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

used for these calculations are described in Introduction (p. 818), and instructions for setting up and solving a solidification/melting problem are provided in Modeling Solidification and Melting of the User's Guide.

16.2. Limitations

As mentioned in Overview (p. 817), the formulation in Ansys Fluent can be used to model the solidification/melting of pure materials, as well as alloys. In the case of alloys, Ansys Fluent offers two rules to determine the liquid fraction versus temperature relationship, namely the linear Lever rule and the nonlinear Scheil rule.

The following limitations apply to the solidification/melting model in Ansys Fluent:

- The solidification/melting model can be used only with the pressure-based solver; it is not available with the density-based solvers.
- The solidification/melting model cannot be used for compressible flows.
- Of the general multiphase models (VOF, mixture, and Eulerian), only the VOF model can be used with the solidification/melting model.
- With the exception of species diffusivities, you cannot specify separate material properties for the solid and liquid materials through the user interface. However, if needed, separate solid and liquid properties can be specified with user-defined functions using the DEFINE_PROPERTY macro. For details on using the DEFINE_PROPERTY macro, refer to DEFINE_PROPERTY UDFs in the Fluent Customization Manual.
- When using the solidification/melting model in conjunction with modeling species transport with reactions, there is no mechanism to restrict the reactions to only the liquid region; that is, the reactions are solved everywhere.

16.3. Introduction

An enthalpy-porosity technique [673] (p. 1092), [675] (p. 1092), [676] (p. 1092) is used in Ansys Fluent for modeling the solidification/melting process. In this technique, the melt interface is not tracked explicitly. Instead, a quantity called the liquid fraction, which indicates the fraction of the cell volume that is in liquid form, is associated with each cell in the domain. The liquid fraction is computed at each iteration, based on an enthalpy balance.

The mushy zone is a region in which the liquid fraction lies between 0 and 1. The mushy zone is modeled as a "pseudo" porous medium in which the porosity decreases from 1 to 0 as the material solidifies. When the material has fully solidified in a cell, the porosity becomes zero and hence the velocities also drop to zero.

In this section, an overview of the solidification/melting theory is given. Refer to Voller and Prakash [676] (p. 1092) for details on the enthalpy-porosity method.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

16.4. Energy Equation

The enthalpy of the material is computed as the sum of the sensible enthalpy, h, and the latent heat, ΔH:

$$H = h + \Delta H \tag{16.1}$$

where

$$h = h_{ref} + \int_{T_{ref}}^{T} c_p dT \tag{16.2}$$

and

h_{ref} = reference enthalpy

T_{ref} = reference temperature

c_p = specific heat at constant pressure

The liquid fraction, β, can be defined as

$$\begin{aligned} &\beta = 0 && \text{if} \quad T < T_{solidus} \\ &\beta = 1 && \text{if} \quad T > T_{liquidus} \\ &\beta = \frac{T - T_{solidus}}{T_{liquidus} - T_{solidus}} && \text{if} \quad T_{solidus} < T < T_{liquidus} \end{aligned} \tag{16.3}$$

The latent heat content can now be written in terms of the latent heat of the material, L:

$$\Delta H = \beta L \tag{16.4}$$

The latent heat content can vary between zero (for a solid) and L (for a liquid).

For solidification/melting problems, the energy equation is written as

$$\frac{\partial}{\partial t}(\rho H) + \nabla \cdot (\rho \vec{v} H) = \nabla \cdot (k \nabla T) + S \tag{16.5}$$

where

H = enthalpy (see Equation 16.1 (p. 819))

ρ = density

$\vec{v}$ = fluid velocity

S = source term

The solution for temperature is essentially an iteration between the energy equation (Equation 16.5 (p. 819)) and the liquid fraction equation (Equation 16.3 (p. 819)). Directly using Equation 16.3 (p. 819) [677] (p. 1092) to update the liquid fraction usually results in poor convergence of the energy equation. In Ansys Fluent, the method suggested by Voller and Swaminathan is used to update the liquid fraction. For pure metals, where $T_{solidus}$ and $T_{liquidus}$ are equal, a method based on specific heat, given by Voller and Prakash [676] (p. 1092), is used instead.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

16.5. Momentum Equations

The enthalpy-porosity technique treats the mushy region (partially solidified region) as a porous medium. The porosity in each cell is set equal to the liquid fraction in that cell. In fully solidified regions, the porosity is equal to zero, which extinguishes the velocities in these regions. The momentum sink due to the reduced porosity in the mushy zone takes the following form:

$$S=\frac{\left(1-\beta\right)^{2}}{\left(\beta^{3}+\varepsilon\right)}A_{mush}\left(\vec{v}-\vec{v}_{p}\right) \tag{16.6}$$

where β is the liquid volume fraction, ε is a small number (0.001) to prevent division by zero, A_{mush} is the mushy zone constant, and $\vec{v}_p$ is the solid velocity due to the pulling of solidified material out of the domain (also referred to as the pull velocity).

The mushy zone constant measures the amplitude of the damping; the higher this value, the steeper the transition of the velocity of the material to zero as it solidifies. Very large values may cause the solution to oscillate.

The pull velocity is included to account for the movement of the solidified material as it is continuously withdrawn from the domain in continuous casting processes. The presence of this term in Equation 16.6 (p. 820) allows newly solidified material to move at the pull velocity. If solidified material is not being pulled from the domain, $\vec{v}_p=0$. More details about the pull velocity are provided in Pull Velocity for Continuous Casting (p. 823).

16.6. Turbulence Equations

Sinks are added to all of the turbulence equations in the mushy and solidified zones to account for the presence of solid matter. The sink term is very similar to the momentum sink term (Equation 16.6 (p. 820)):

$$S=\frac{\left(1-\beta\right)^{2}}{\left(\beta^{3}+\varepsilon\right)}A_{mush}\varphi \tag{16.7}$$

where φ represents the turbulence quantity being solved (k, ε, ω, and so on), and the mushy zone constant, A_{mush}, is the same as the one used in Equation 16.6 (p. 820).

16.7. Species Equations

For solidification and melting of a pure substance, phase change occurs at a distinct melting temperature, T_{melt}. For a multicomponent mixture, however, a mushy freeze/melt zone exists between a lower solidus and an upper liquidus temperature. When a multicomponent liquid solidifies, solutes diffuse from the solid phase into the liquid phase. This effect is quantified by the partition coefficient of solute i, denoted K_i, which is the ratio of the mass fraction in the solid to that in the liquid at the interface.

Ansys Fluent computes the solidus and liquidus temperatures in a species mixture as,

$$T_{solidus} = T_{melt}+\sum_{solutes} m_i Y_i / K_i \tag{16.8}$$

$$T_{liquidus} = T_{melt}+\sum_{solutes} m_i Y_i \tag{16.9}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

where K_i is the partition coefficient of solute i, Y_i is the mass fraction of solute i, and m_i is the slope of the liquidus surface with respect to Y_i. If the value of the mass fraction Y_i exceeds the value of the eutectic mass fraction $Y_{i,\mathrm{Eut}}$, then Y_i is clipped to $Y_{i,\mathrm{Eut}}$ when calculating the liquidus and solidus temperatures. It is assumed that the last species material of the mixture is the solvent and that the other species are the solutes.

Ansys Fluent expects that you will input a negative value for the liquidus slope of species i (m_i). If you input a positive slope for m_i, Ansys Fluent will disregard your input and instead calculate it using the Eutectic temperature T_{Eut} and the Eutectic mass fraction $Y_{i,\mathrm{Eut}}$:

$$m_i = \frac{T_{Eut} - T_{melt}}{Y_{i,Eut}} \tag{16.10}$$

Updating the liquid fraction via Equation 16.3 (p. 819) can cause numerical errors and convergence difficulties in multicomponent mixtures. Instead, the liquid fraction is updated as,

$$\beta^{n+1} = \beta^n - \lambda \frac{a_p\left(T - T^*\right)\Delta t}{\rho V L - a_p \Delta t \frac{\partial T^*}{\partial \beta}} \tag{16.11}$$

where the superscript n indicates the iteration number, λ is a relaxation factor with a default value of 0.9, a_p is the cell matrix coefficient, Δt is the time step, ρ is the current density, V is the cell volume, T is the current cell temperature and T^* is the interface temperature.

Ansys Fluent offers two models for species segregation at the micro-scale, namely the Lever rule and the Scheil rule. The former assumes infinite diffusion of the solute species in the solid, and the latter assumes zero diffusion. For the Lever rule, the interface temperature T^*, is calculated as:

$$T^* = T_{\mathrm{melt}} + \sum_{i=0}^{N_s - 1} m_i \left(\frac{Y_i}{K_i + \beta\left(1 - K_i\right)} \right) \tag{16.12}$$

where N_s is the number of species.

The Scheil rule evaluates T^* as:

$$T^* = T_{\mathrm{melt}} + \sum_{i=0}^{N_s - 1} m_i Y_i \beta^{K_i - 1} \tag{16.13}$$

For information about how back diffusion (that is, a finite amount of diffusion of the solute species) can be incorporated into the formulation, see the section that follows.

For the Lever rule, species transport equations are solved for the total mass fraction of species i, Y_i:

$$\frac{\partial}{\partial t}\left(\rho Y_i\right) + \nabla \cdot \left(\rho \left[\beta \vec{v}_{liq} Y_{i,liq} + \left(1 - \beta\right) \vec{v}_p Y_{i,sol} \right] \right) = -\nabla \cdot \vec{J}_i + R_i \tag{16.14}$$

where R_i is the reaction rate and $\vec{J}_i$ is given by

$$\vec{J}_i = -\rho \left[\beta D_{i,m,liq} \nabla Y_{i,liq} + \left(1 - \beta\right) D_{i,m,sol} \nabla Y_{i,sol} \right] \tag{16.15}$$

$\vec{v}_{liq}$ is the velocity of the liquid and $\vec{v}_p$ is the solid (pull) velocity. $\vec{v}_p$ is set to zero if pull velocities are not included in the solution. The liquid velocity can be found from the average velocity (as determined by the flow equation) as

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$\vec{v}_{liq}=\frac{\left(\vec{v}-\vec{v}_p\left(1-\beta\right)\right)}{\beta} \tag{16.16}$$

The liquid ($Y_{i,liq}$) and solid ($Y_{i,sol}$) mass fractions are related to each other by the partition coefficient K_i:

$$Y_{i,sol}=K_iY_{i,liq} \tag{16.17}$$

When the Scheil model is selected, Ansys Fluent solves for $Y_{i,liq}$ as the dependent variable [674] (p. 1092):

$$\begin{aligned}\frac{\partial}{\partial t}\left(\rho Y_{i,liq}\right)+\nabla\cdot\left(\rho\left[\beta\vec{v}_{liq}Y_{i,liq}+\left(1-\beta\right)\vec{v}_pY_{i,sol}\right]\right) &= R_i\\ &+\nabla\cdot\left(\rho\beta D_{i,m,liq}\nabla Y_{i,liq}\right)\\ &-K_iY_{i,liq}\frac{\partial}{\partial t}\left(\rho\left(1-\beta\right)\right)\\ &+\frac{\partial}{\partial t}\left(\rho\left(1-\beta\right)Y_{i,liq}\right)\end{aligned} \tag{16.18}$$

16.8. Back Diffusion

The Scheil rule assumes zero solute diffusion in the solid zone, while the Lever rule assumes infinite solute diffusion in the solid. Back diffusion models a finite solid diffusion at the micro level using a non-dimensional parameter (denoted by γ) between 0 (no diffusion, as in the Scheil rule) and 1 (infinite diffusion, as in the Lever Rule). γ typically depends on the solidification conditions such as the solid diffusivity, the local solidification time, and the secondary dendritic arm spacing. The interface temperature is calculated as follows:

$$T^*=T_{\text{melt}}+\sum_{i=0}^{N_s-1}m_iY_i\left(1-\left(1-\gamma K_i\right)\left(1-\beta\right)\right)^{\frac{K_i-1}{1-\gamma K_i}} \tag{16.19}$$

When back diffusion is included with the Scheil rule formulation, the species transport equation is solved with the liquid species mass fraction as the dependent variable:

$$\begin{aligned}\frac{\partial\left(\rho Y_{i,\text{liq}}\right)}{\partial t}+\nabla\cdot\left(\rho\left[\beta\vec{v}Y_{i,\text{liq}}+\left(1-\beta\right)\vec{v}_pY_{i,\text{sol}}\right]\right) &= R_i\\ &+\nabla\cdot\left(\rho\beta D_{i,m,\text{liq}}\nabla Y_{i,\text{liq}}\right)\\ &-K_iY_{i,\text{liq}}\frac{\partial\left(\rho\left(1-\beta\right)\right)}{\partial t}\\ &+\frac{\partial\left(\rho\left(1-\beta\right)Y_{i,\text{liq}}\right)}{\partial t}\\ &-\gamma K_i\left(1-\beta\right)\frac{\partial\left(\rho Y_{i,\text{liq}}\right)}{\partial t}\end{aligned} \tag{16.20}$$

The total species mass fraction and liquid species mass fraction are related as

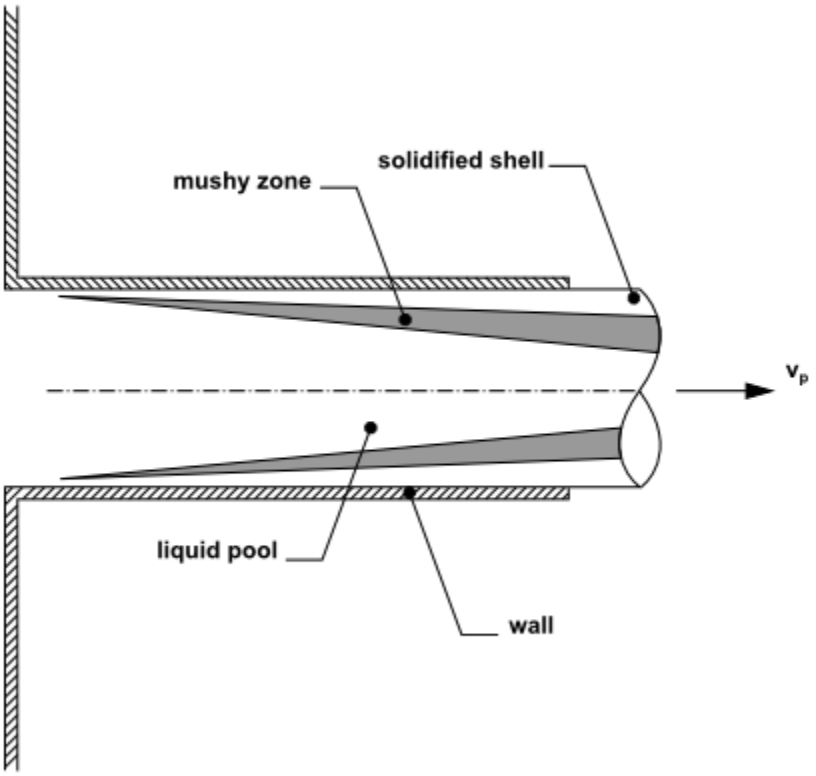
$$Y_{i,\text{sol}}=K_iY_i\left[1-\left(1-\gamma K_i\right)\left(1-\beta\right)\right]^{\frac{K_i-1}{1-\gamma K_i}}=K_iY_{i,\text{liq}} \tag{16.21}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

16.9. Pull Velocity for Continuous Casting

In continuous casting processes, the solidified matter is usually continuously pulled out from the computational domain, as shown in Figure 16.1: "Pulling" a Solid in Continuous Casting (p. 823). Consequently, the solid material will have a finite velocity that must be accounted for in the enthalpy-porosity technique.

Figure 16.1: "Pulling" a Solid in Continuous Casting



As mentioned in Momentum Equations (p. 820), the enthalpy-porosity approach treats the solid-liquid mushy zone as a porous medium with porosity equal to the liquid fraction. A suitable sink term is added in the momentum equation to account for the pressure drop due to the porous structure of the mushy zone. For continuous casting applications, the relative velocity between the molten liquid and the solid is used in the momentum sink term (Equation 16.6 (p. 820)) rather than the absolute velocity of the liquid.

The exact computation of the pull velocity for the solid material is dependent on Young's modulus and Poisson's ratio of the solid and the forces acting on it. Ansys Fluent uses a Laplacian equation to approximate the pull velocities in the solid region based on the velocities at the boundaries of the solidified region:

$$\nabla^2 \vec{v}_p = 0 \tag{16.22}$$

Ansys Fluent uses the following boundary conditions when computing the pull velocities:

- At a velocity inlet, a stationary wall, or a moving wall, the specified velocity is used.
- At all other boundaries (including the liquid-solid interface between the liquid and solidified material), a zero-gradient velocity is used.

The pull velocities are computed only in the solid region.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Note that Ansys Fluent can also use a specified constant value or custom field function for the pull velocity, instead of computing it. See Procedures for Modeling Continuous Casting of the User's Guide for details.

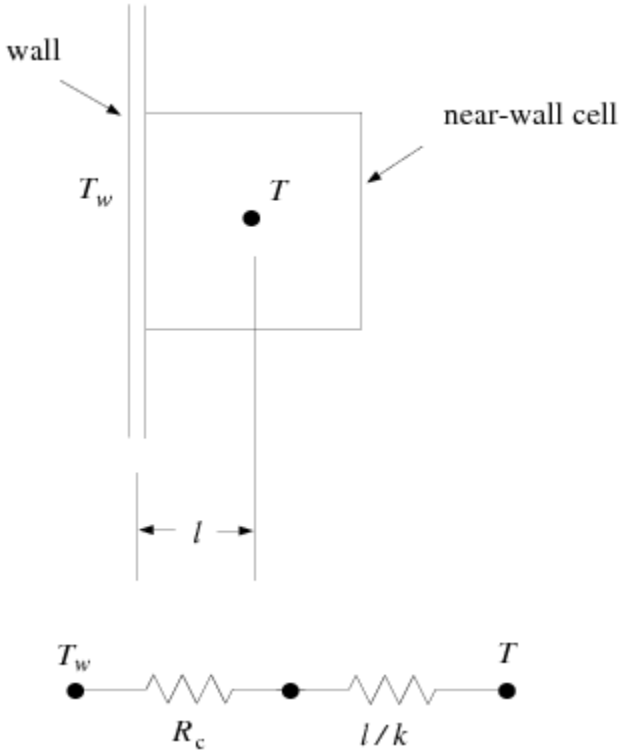
16.10. Contact Resistance at Walls

Ansys Fluent's solidification/melting model can account for the presence of an air gap between the walls and the solidified material, using an additional heat transfer resistance between walls and cells with liquid fractions less than 1. This contact resistance is accounted for by modifying the conductivity of the fluid near the wall. Thus, the wall heat flux, as shown in Figure 16.2: Circuit for Contact Resistance (p. 824), is written as

$$q=\frac{\left(T-T_w\right)}{\left(l/\ k+R_c\left(1-\beta\right)\right)} \tag{16.23}$$

where T, T_w, and l are defined in Figure 16.2: Circuit for Contact Resistance (p. 824), k is the thermal conductivity of the fluid, β is the liquid volume fraction, and R_c is the contact resistance, which has the same units as the inverse of the heat transfer coefficient.

Figure 16.2: Circuit for Contact Resistance



16.11. Thermal and Solutal Buoyancy

When multiple species are involved in the solidification/melting process, the density of the liquid pool varies with temperature as well as species composition. In the presence of a gravitational field, buoyancy will be induced by two mechanisms:

- **Thermal buoyancy**: is the gravitational force due to the variation of density with temperature.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

- **Solutal buoyancy**: is the gravitational force due to the variation of density with the change in the species composition of the melt. In multi-component solidification problems, the solutes are continuously rejected near the liquid solid interface, leading to the enrichment of the liquid pool with the solutes. This enrichment of the solutes causes a concentration gradient in the immediate vicinity of the solid-liquid interface, and consequently a density gradient.

Modeling of thermal and solutal buoyancy is often important for the accurate prediction of the overall solidification behavior. The flow due to thermal buoyancy tends to promote mixing and smooths out temperature gradients. Therefore, excluding this term can lead to inaccurate predictions of solidification time. Solutal buoyancy moves the enriched liquid away from the liquid-solid interface and replaces it with the far field nominal composition liquid. Excluding solutal buoyancy can lead to an over prediction of segregation patterns for multi-component materials.

For multi-component solidification problems, the buoyancy induced flows are modeled in Ansys Fluent using the Boussinesq approach. The thermal buoyancy is calculated as described in Natural Convection and Buoyancy-Driven Flows Theory (p. 176). The body forces due to solutal buoyancy are calculated using a similar approach:

$$\vec{F}_s = \rho_{ref}\vec{g}\sum_{i=0}^{N_s}\beta_{s,i}\left(y_{l,i} - y_{ref,i}\right) \tag{16.24}$$

where $y_{l,i}$ is the mass fraction of the ith solute in the liquid phase and $y_{ref,i}$ is the reference mass fraction of the i^{th} species. $\beta_{s,i}$ is the solutal expansion coefficient of the i^{th} species and N_s is the total number of solute species, g is the gravity vector, and ρ_{ref} is the reference density.

The total body force is the sum of the solutal and thermal buoyancy, which can have similar or opposite signs, and their relative importance can change significantly during the course of the simulation.

Refer to Modeling Thermal and Solutal Buoyancy in the User's Guide to learn how to include thermal and solutal buoyancy.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information
of ANSYS, Inc. and its subsidiaries and affiliates.

Chapter 17: The Structural Model for Intrinsic Fluid-Structure Interaction (FSI)

This chapter presents the theoretical basis of the linear and nonlinear elasticity structural model, which can be used to perform a flow analysis involving the prediction of stresses in a solid affected by a fluid flow. This feature is available for 2D planar and 3D meshes, but not for 2D axisymmetric meshes. In terms of physics, the feature is limited to linear isothermal and isotropic elasticity.

Although solving an elastic case on its own can be interesting, the real added value of the structural model is the ability to perform a fluid-structure interaction (FSI) simulation completely within a Fluent session. This is referred to as "intrinsic FSI", as all of the structural calculations are performed by Fluent; note that you can also use extrinsic solvers for FSI problems, incorporating the structural data into your Fluent fluid simulation through mapping or system coupling. The flow fields will apply forces on the interface boundary with the solid, which will deform; the resulting displacements of the solid domain will in turn lead to a deformation of the fluid domain.

The problem of linear or nonlinear elasticity is of the elliptic type, that is, the behavior is mainly dictated by boundary values (instead of initial values). Therefore, it is rather natural to use a finite element method for solving the stress and deformations in the solid body. On the fluid side of the FSI boundary, the physics is often dictated by transport (at least when the Reynolds number is sufficiently large), and so a finite volume method is appropriate. The art consists of establishing a communication between both of these regions / methods.

Theoretical information about the structural model is presented in the following sections:

17.1. Limitations

17.2. The FSI Model

17.3. Intrinsic FSI

17.4. Linear Elasticity

17.5. Nonlinear Elasticity

17.6. Thermoelasticity Model

17.7. Coupling of the Structural Model with the Battery Swelling Model

For information about using the structural model and simulation setup, see Modeling Fluid-Structure Interaction (FSI) Within Fluent in the User's Guide.

17.1. Limitations

The following limitations apply to intrinsic FSI simulations:

- It is available for 3D models as well as 2D planar stress models (2D planar strain models are not considered), however, 2D axisymmetric and axisymmetric swirl are supported only by the Linear Elasticity model.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

- Non-isothermal effects are not included in the solid body—that is, thermal dilatation is omitted.
- Solid properties (Young's modulus and Poisson's ratio) must be constant.
- Only linear interpolation functions are used.
- Polyhedral elements are not supported. Only tetrahedral, hexahedral, wedges, and pyramid elements can be used for 3D cases, and only triangular and quadrilateral elements can be used for 2D cases.
- The default explicit mesh updating used for two-way FSI simulations is unstable under certain conditions (such as when the fluid and solid densities are comparable or when large deformations are considered). In such situations, implicit dynamic mesh update is recommended.
- The linear elasticity model assumes that the stress loading does not exceed the yield strength of the solid material.

17.2. The FSI Model

In an FSI context, the flow applies forces on the interface boundary with the solid, which will deform; the resulting displacements of the solid domain will in turn lead to a deformation of the fluid domain. The communication between the fluid and the solid occurs in terms of forces from the fluid to the solid and in terms of displacements from the solid to the fluid. When the displacements undergone by the solid domain are small, feedback from the solid deformation to the fluid domain can be neglected: this is referred to as one-way FSI simulation. When the displacements undergone by the solid domain are no longer negligible, feedback of the solid deformation onto the fluid domain is necessary: this is referred to as two-way FSI simulation.

17.3. Intrinsic FSI

Intrinsic FSI is used to designate a simulation of fluid-structure interaction in which the same solver is used for solving simultaneously the flow governing equations and the solid deformations. In an intrinsic FSI context, fluid and solid domains share a common interface, whose discretization is conformal. The displacement of the solid domain will result from the continuous forces generated by the fluid; these forces are of course a priori unknown, and are part of the calculation.

If the expected displacements of the solid region are negligible, it will be sufficient to consider a one-way FSI simulation. A one-way FSI simulation can be steady or transient; this will be dictated by the settings selected for the fluid flow simulation.

A two-way FSI approach should be used when the displacements calculated on the solid domain affect the fluid domain, such as in the following examples: a high pressure flow around a flexible aluminum flap, or a cantilevered thin plate that is initially deformed and then allowed to freely oscillate in a fluid. For such cases, the fluid domain will deform in accordance with the displacements calculated on the solid domain; of course, these displacements are also a priori unknown and are part of the entire calculation. In a two-way FSI context, the physics and the numerics involved on both sides of the fluid-solid interface differ; however, it is necessary to ensure the continuity of the displacements and of the velocity, as well as the equilibrium of stresses. Two-way FSI is solved by means of a transient scheme, and explicit mesh updating is used by default, such that the dynamic mesh is only updated at the beginning of a time step; for cases with strong FSI coupling, you should enable the implicit update scheme instead, so that it is updated during a time step.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

It is important to note that small displacements are considered within the context of linear elasticity. Therefore, although the solid body also deforms, the calculation of the displacement governing equations on the solid body will be performed on the basis of the initial coordinates.

17.4. Linear Elasticity

17.4.1. Equations

17.4.1.1. Linear Isotropic and Isothermal Elasticity

Consider a solid body Σ surrounded by a boundary Ω. In general, volume forces are applied on Σ while surface forces and/or displacements are imposed on Ω. Let $\underline{u}$ denote the displacement vector defined on Σ. which results from the forces and displacements imposed on Ω . From $\underline{u}$. the deformation tensor $\underline{\underline{\varepsilon}}$ can be calculated as the symmetric part of the gradient of $\underline{u}$:

$$\underline{\underline{\varepsilon}}=\frac{1}{2}\left(\underline{\underline{\nabla u}}+\left(\underline{\underline{\nabla u}}\right)^{t}\right) \tag{17.1}$$

In the present context of linear isothermal and isotropic elasticity, the constitutive equation of the stresses $\underline{\underline{\sigma}}$ is given by

$$\underline{\underline{\sigma}}=\frac{E}{1+\nu}\left(\frac{\nu}{1-2\nu}\mathrm{tr}\left(\underline{\underline{\varepsilon}}\right)\underline{\underline{I}}+\underline{\underline{\varepsilon}}\right) \tag{17.2}$$

where $\underline{\underline{I}}$ is the unit tensor. This equation involves two physical parameters: the Young's modulus E and the Poisson's ratio ν.

When introducing this expression for the stresses $\underline{\underline{\sigma}}$ into the momentum equation, and after substituting $\underline{\underline{\varepsilon}}$ by its definition, a second-order elliptic equation is eventually obtained for the displacement $\underline{u}$. After substitution, the momentum equation can be written as:

$$0=-\rho\underline{a}+\nabla\cdot\underline{\underline{\sigma}}=-\rho\underline{a}+\frac{E}{1+\nu}\nabla\cdot\left(\frac{\nu}{1-2\nu}\mathrm{tr}\left(\underline{\underline{\nabla u}}\right)\underline{\underline{I}}+\frac{1}{2}\left(\underline{\underline{\nabla u}}+\left(\underline{\underline{\nabla u}}\right)^{t}\right)\right) \tag{17.3}$$

where ρ is the density of the solid and $\underline{a}$ is the acceleration. This equation requires that boundary conditions be given in terms of displacement or in terms of force. So, on a boundary where the solid body is fixed, a zero displacement will be assigned, whereas nodal force or continuous forces can be applied on freestanding walls. In an FSI simulation, on a boundary where the solid is in contact with a surrounding fluid, continuous forces originating from the fluid will be assigned as boundary conditions.

17.4.1.2. Evaluation of the von Mises Stress

The von Mises stress is invoked in the von Mises criterion used for anticipating the occurrence of yielding of ductile materials, such as metals. It is evaluated as:

$$\sigma_{e}=\left(\frac{1}{2}\left[\left(\sigma_{xx}-\sigma_{yy}\right)^{2}+\left(\sigma_{yy}-\sigma_{zz}\right)^{2}+\left(\sigma_{zz}-\sigma_{xx}\right)^{2}+6\left(\sigma_{xy}^{2}+\sigma_{yz}^{2}+\sigma_{xz}^{2}\right)\right]\right)^{\frac{1}{2}} \tag{17.4}$$

The equation is algebraic, so that the von Mises stress can directly be evaluated from the previously calculated stress components.

For 2D planar stress state, the relationship becomes:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$\sigma_e=\left(\frac{1}{2}\left[(\sigma_{xx}-\sigma_{yy})^2+\sigma_{xx}^2+\sigma_{yy}^2+6\sigma_{xy}^2)\right]\right)^{\frac{1}{2}}=\left(\sigma_{xx}^2+\sigma_{yy}^2-\sigma_{xx}\sigma_{yy}+3\sigma_{xy}^2\right)^{\frac{1}{2}} \quad (17.5)$$

The von Mises stress can be invoked to linear and non-linear elasticity cases. It is originally developed for ductile materials, such as metals. The application towards other solid materials is subject to validation and common sense.

17.4.2. Finite Element Representation

With the exception of a few simple cases, Equation 17.3 (p. 829) cannot be solved analytically, and so a numerical scheme must be used. As described previously, the problem of linear elasticity is of the elliptic type, that is, the behavior is mainly dictated by boundary values (instead of initial values). Therefore, it is rather natural to use a finite element method for solving the stress and deformations in the solid body. In the finite element context, the structure being analyzed is discretized (approximated) as an assembly of *N* discrete regions (elements) connected at a finite number (*M*) of points, or nodes. These elements are triangles or quadrilaterals for 2D cases, while tetrahedra, hexahedra, wedges or pyramids are used for 3D cases.

Displacement unknowns are defined at nodes and connected with appropriate interpolation functions at the level of each element. These interpolation functions are continuous at the level of each element, as well as the first derivative; they are selected in such a way that continuity is also guaranteed at the interface between adjacent elements. In many applications, continuous piecewise linear interpolation functions are considered. There are as many such interpolation functions ϕ_i as unknown nodal displacement vectors u_i . and in particular each interpolation function is such that ϕ_i =1 at the node i and =0 at the other nodes.

17.4.2.1. Construction of the Matrix of the System

In the first instance, let us omit the acceleration term in Equation 17.3 (p. 829) . Let $\underline{\tilde{u}}$ be the approximation for the displacement field, described in terms of nodal values u_i and interpolation functions ϕ_i . The interpolation functions and the approximation $\underline{\tilde{u}}$ are used for transforming the partial differential Equation 17.3 (p. 829) into a linear algebraic system of *M* vector equations with *M* vector unknowns. For this, the following procedure is applied:

- *M* equations are created by multiplying Equation 17.3 (p. 829) with the *M* interpolation functions.
- The part integration rule and Green's theorem are applied to decrease the level of the derivative by one order and to easily embed force boundary conditions into the system.
- Eventually, the selected approximation for the displacement field described in terms of nodal values u_i and interpolation functions is introduced into the system.

The discretized form of Equation 17.3 (p. 829) now becomes

$$0=-\int_{\Omega}\phi_i\underline{f}_{\Omega}d\Omega+\int_{\Sigma}\frac{E}{1+\nu}\nabla\phi_i\cdot\left(\frac{\nu}{1-2\nu}\mathrm{tr}\left(\underline{\underline{\nabla\tilde{u}}}\right)\underline{\underline{I}}+\frac{1}{2}\left(\underline{\underline{\nabla\tilde{u}}}+\left(\underline{\underline{\nabla\tilde{u}}}\right)^t\right)\right)d\Sigma \quad (17.6)$$

where $\underline{f}_{\Omega}$ is the vector of external forces applied on the boundary Ω . For the sake of facility, Equation 17.6 (p. 830) can be rearranged as

$$[K]\{u\}=\{f\} \quad (17.7)$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

In Equation 17.7 (p. 830) . $[K]$ is the total stiffness matrix, $\{u\}$ is the vector of nodal displacement unknowns u_i . while $\{f\}$ results from the external (continuous or nodal) forces applied on the boundary of the solid body. In particular, in an FSI simulation, they incorporate the forces originating from the fluid motion. The additivity property allows the integration in Equation 17.6 (p. 830) to be performed at the level of each individual element in order to obtain a local stiffness matrix, which is subsequently assembled into the total stiffness matrix $[K]$ of the linear system. After the assembly of the total stiffness matrix is completed, displacement boundary conditions are applied by removing the corresponding unknowns and equations from the system. These methods are well documented (for example, see Zienkiewicz [729] (p. 1095)). For solving the system in the present context, an iterative solver is used.

17.4.2.2. Dynamic Structural Systems

In linear structural dynamics systems, the internal load is linearly proportional to the nodal displacement, and the structural stiffness matrix remains constant. However, inertia terms should be taken into account into the momentum equation, as well as a possible term for structural damping. The discretized Equation 17.7 (p. 830) now becomes

$$[M]\{\ddot{u}(t)\}+[C]\{\dot{u}(t)\}+[K]\{u(t)\}=\{f(t)\} \tag{17.8}$$

where $[M]$. $[C]$. and $[K]$ are the structural mass matrix, the structural damping matrix, and the stiffness matrix, respectively, while $\{\ddot{u}(t)\}$. $\{\dot{u}(t)\}$. and $\{u(t)\}$ are the vectors of nodal accelerations, velocities, and displacements, respectively, and $\{f(t)\}$ is the applied load vector. In a linear structural dynamics system, the matrices are constant, while the vectors are a function of time. Presently, the structural damping matrix $[C]$ is not considered, however, for the sake of generality, it is kept in the equations.

Among direct time integration methods for numerically solving the finite element semi-discrete equation of motion given Equation 17.8 (p. 831) . several methods are available in literature; two such methods are the Newmark method (Newmark [472] (p. 1080)) and the backward Euler method, which are available in Fluent and are described in the sections that follow.

17.4.2.2.1. The Newmark Method

The Newmark family of time integration algorithms (Newmark [472] (p. 1080)) is one of the most popular time integration methods as a single-step algorithm. As noted by Hughes [260] (p. 1068). the semi-discrete Equation 17.8 (p. 831) can be rewritten as:

$$[M]\{\ddot{u}_{n+1}\}+[C]\{\dot{u}_{n+1}\}+[K]\{u_{n+1}\}=\{f_{n+1}\} \tag{17.9}$$

where $\{\ddot{u}_{n+1}\}$. $\{\dot{u}_{n+1}\}$. and $\{u_{n+1}\}$ are the vectors of nodal accelerations, velocities, and displacements at time t_{n+1} . respectively, while $\{f_{n+1}\}$ is the load vector applied at time t_{n+1} . In addition to Equation 17.9 (p. 831) . the Newmark family of time integration algorithms requires the displacement and velocity to be updated as

$$\{\dot{u}_{n+1}\}=\{\dot{u}_n\}+\left[(1-\delta)\{\ddot{u}_n\}+\delta\{\ddot{u}_{n+1}\}\right]\Delta t \tag{17.10}$$

$$\{u_{n+1}\}=\{u_n\}+\{\dot{u}_n\}\Delta t+\left[\left(\tfrac{1}{2}-\alpha\right)\{\ddot{u}_n\}+\alpha\{\ddot{u}_{n+1}\}\right]\Delta t^2 \tag{17.11}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

where α and δ are Newmark's integration parameters and $\{\ddot{u}_n\}$. $\{\dot{u}_n\}$. and $\{u_n\}$ are the vectors of nodal accelerations, velocities, and displacements at time t_n . respectively, while $\Delta t = t_{n+1} - t_n$ is the time step. By making use of the three algebraic equations (Equation 17.9 (p. 831) . Equation 17.10 (p. 831) . and Equation 17.11 (p. 831)), a single-step time integrator in terms of the unknowns $\{u_{n+1}\}$ and the three known quantities can be written as

$$\begin{aligned}&\left(\alpha_0[M]+\alpha_1[C]+[K]\right)\{u_{n+1}\}\\&\quad=\{f_{n+1}\}+[M]\left(\alpha_0\{u_n\}+\alpha_2\{\dot{u}_n\}+\alpha_3\{\ddot{u}_n\}\right)+[C]\left(\alpha_1\{u_n\}+\alpha_4\{\dot{u}_n\}+\alpha_5\{\ddot{u}_n\}\right)\end{aligned} \tag{17.12}$$

where the several coefficients are given by $\alpha_0 = 1/\alpha\Delta t^2$. $\alpha_1 = \delta/\alpha\Delta t$. $\alpha_2 = 1/\alpha\Delta t$. $\alpha_3 = 1/\Delta t - 1$. $\alpha_4 = \delta/\alpha - 1$. and $\alpha_5 = \Delta t/2(\delta/\alpha - 2)$.

The calculation procedure is as follows. First,the vector of nodal displacements $\{u_{n+1}\}$ is calculated using Equation 17.12 (p. 832) . Then, the program computes the vectors of nodal velocities $\{\dot{u}_{n+1}\}$ and nodal accelerations $\{\ddot{u}_{n+1}\}$ with the following relationships:

$$\{\dot{u}_{n+1}\} = \alpha_1\left(\{u_{n+1}\} - \{u_n\}\right) - \alpha_4\{\dot{u}_n\} - \alpha_5\{\ddot{u}_n\} \tag{17.13}$$

$$\{\ddot{u}_{n+1}\} = \alpha_0\left(\{u_{n+1}\} - \{u_n\}\right) - \alpha_2\{\dot{u}_n\} - \alpha_3\{\ddot{u}_n\} \tag{17.14}$$

The most important factors in choosing an appropriate time integration scheme for the finite element semi-discrete equation of motion (Equation 17.7 (p. 830)) are accuracy, stability, and dissipation. In conditionally stable time integration algorithms, stability is affected by the selected time step; whereas in unconditionally stable time integration algorithms, a time step can be selected independently of stability considerations.

In the Newmark method, the amount of numerical algorithm dissipation can be controlled by one of the Newmark's parameters (δ) as follows:

$$\begin{aligned}&\delta \geq \tfrac{1}{2}\\&\alpha \geq \tfrac{1}{4}\left(\tfrac{1}{2}+\delta\right)^2\end{aligned} \tag{17.15}$$

With the Newmark parameters satisfying the above conditions, the Newmark family of methods is unconditionally stable (Hughes [260] (p. 1068)). By introducing the amplitude decay factor γ. the above conditions can be written as:

$$\begin{aligned}&\delta = \tfrac{1}{2}+\gamma\\&\alpha = \tfrac{1}{4}\left(1+\gamma\right)^2\\&\gamma \geq 0\end{aligned} \tag{17.16}$$

Consequently, Fluent provides you with the Newmark integration procedure, which is unconditionally stable through the input of the amplitude decay factor γ.

17.4.2.2.2. Backward Euler Method

The backward Euler method requires velocity and acceleration to be updated as follows:

$$\{\dot{u}_{n+1}\} = \frac{\{u_{n+1}\} - \{u_n\}}{\Delta t} \tag{17.17}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$\{\ddot{u}_{n+1}\}=\frac{\{\dot{u}_{n+1}\}-\{\dot{u}_n\}}{\Delta t} \tag{17.18}$$

Combining Equation 17.17 (p. 832) and Equation 17.18 (p. 833) with the semi-discrete equation of motion given in Equation 17.9 (p. 831) results in a single-step time integrator in terms of the unknown $\{u_{n+1}\}$ as:

$$\left(\frac{1}{\Delta t^2}[M]+\frac{1}{\Delta t}[C]+[K]\right)\{u_{n+1}\}=\{F^a_{n+1}\}+[M]\left(\frac{\{u_n\}}{\Delta t^2}+\frac{\{\dot{u}_n\}}{\Delta t}\right)+[C]\frac{\{u_n\}}{\Delta t} \tag{17.19}$$

17.4.2.2.3. Rayleigh Damping

Rayleigh damping is available in the structural model for transient intrinsic fluid-structure interaction (FSI) cases. As described in The Newmark Method (p. 831) . the final form of the equation of transient motion can be written as

$$\begin{aligned}&\left(\alpha_0[M]+\alpha_1[C]+[K]\right)\{u_{n+1}\}\\&\quad=\left\{f_{n+1}\right\}+[M]\left(\alpha_0\{u_n\}+\alpha_2\{\dot{u}_n\}+\alpha_3\{\ddot{u}_n\}\right)+[C]\left(\alpha_1\{u_n\}+\alpha_4\{\dot{u}_n\}+\alpha_5\{\ddot{u}_n\}\right)\end{aligned} \tag{17.20}$$

where the several coefficients are given by $\alpha_0=1/\alpha\Delta t^2$. $\alpha_1=\delta/\alpha\Delta t$. $\alpha_2=1/\alpha\Delta t$. $\alpha_3=1/\Delta t-1$. $\alpha_4=\delta/\alpha-1$. and $\alpha_5=\Delta t/2\left(\delta/\alpha-2\right)$.

Additionally, the Rayleigh model for structural damping matrix is as follows:

$$[C]=\alpha_d[M]+\beta_d[K] \tag{17.21}$$

where α_d is the mass-proportional damping coefficient, and β_d is the stiffness-proportional damping coefficient.

17.5. Nonlinear Elasticity

When considering the linear elasticity structural model, the linear relationship between strain and displacements was assumed. This assumption is only suitable for as long as small displacement concept is valid. Contrary to that, the large displacement implies a significant change in initial geometry and consequently geometric nonlinearity must be introduced into the solution algorithm (see Bhatti [59] (p. 1056) for additional details).

17.5.1. Finite Element Geometric Nonlinearity

The nonlinear elasticity structural model takes into account geometric nonlinearity in order to simulate large deformation. In the case of large deformation, the reference configuration is based on initial configuration and all quantities are formulated using total Lagrangian formulation. Consequently, a stiffness matrix becomes a function of displacement. One of the main concepts in such formulation is an introduction of deformation gradient formulation that is the main tool in describing the strains in large displacement formulation.

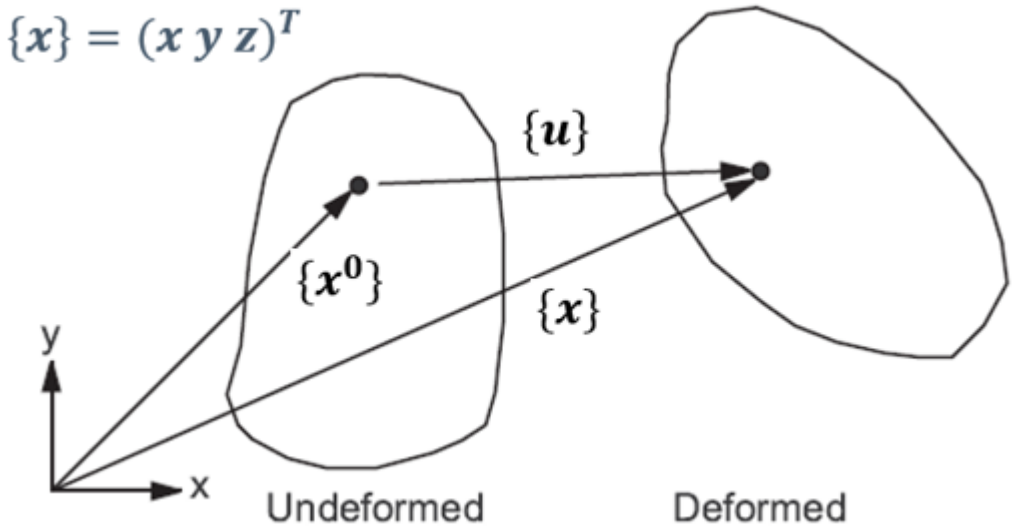
In summary, the nonlinear elasticity structural model is based on following main concepts:

- large strain
- a stiffness matrix function of displacement

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

- a nonlinear relationship between strains and displacements
- a total Lagrangian formulation

The relationship between the initial and deformed state is presented as:



Mathematically, the deformation is expressed through the deformation tensor:

$$\begin{pmatrix} \frac{\partial x}{\partial x^0} & \frac{\partial x}{\partial y^0} & \frac{\partial x}{\partial z^0} \\ \frac{\partial y}{\partial x^0} & \frac{\partial y}{\partial y^0} & \frac{\partial y}{\partial z^0} \\ \frac{\partial z}{\partial x^0} & \frac{\partial z}{\partial y^0} & \frac{\partial z}{\partial z^0} \end{pmatrix} \tag{17.22}$$

17.5.2. Finite Element Nonlinear Discretization

The main task of geometrically nonlinear discretization is to formulate the stiffness matrix. The overall discretization of the solution equation for increments in nodal displacement is based on the tangent element equation:

$$\left(K^C + K^S\right)\Delta u = R_i \tag{17.23}$$

In the previous equation, the total stiffness matrix is made from 2 components: current and stress stiffness matrices. The current stiffness matrix is defined as:

$$K^C = \int B^T F C F^T B dV^0 \tag{17.24}$$

The stress (geometric) stiffness matrix is defined as:

$$K^S = \int B^T S B dV^0 \tag{17.25}$$

where

R_i = the nodal load vector

B = the strain-displacement matrix

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

F = the deformation gradient matrix

C = the constitutive matrix

S = the second Piola-Kirchhoff stress tensor

dV = the element volume

17.5.3. Constitutive Equations

In order to fully close the equations outlined in Finite Element Nonlinear Discretization (p. 834) . it is necessary to define a constitutive equation that defines the relationship between stresses and strains. For large displacement problems, it is assumed that a relationship exists between Green-Lagrange strains and the second Piola-Kirchhoff stresses. Thus, the first-order tensor form can be defined as:

$$C_{ijkl}=\frac{\partial S_{ij}}{\partial \varepsilon_{kl}} \tag{17.26}$$

The vector-matrix form is:

$$S=C\varepsilon \tag{17.27}$$

The second Piola-Kirchhoff stress vector is defined as:

$$S=\left(S_{xx}S_{yy}S_{zz}S_{xy}S_{yz}S_{zx}\right)^{T} \tag{17.28}$$

The Green-Lagrange strain vector is defined as:

$$\varepsilon=\left(\varepsilon_{xx}\varepsilon_{yy}\varepsilon_{zz}\varepsilon_{xy}\varepsilon_{yz}\varepsilon_{zx}\right)^{T} \tag{17.29}$$

The constitutive matrix C would result in a 6x6 matrix tensor. A full derivation of constitutive matrix for large strains is not covered here but we just state that compressible hyper-elastic neo-Hookean material is used as a default for Nonlinear Elasticity. The neo-Hookean model is used as first choice because of its generality and because it is a natural extension of the usual linear isotropic model (Hooke's law) to large deformation.

17.5.4. The Transient Scheme for the Nonlinear System

The default transient scheme for the nonlinear system is the same as that used for linear elasticity: the Newmark Method. The semi-discretized equation of transient motion for the nonlinear system is given as:

$$[M]\{\ddot{u}_{n+1}\}+[C]\{\dot{u}_{n+1}\}+\{F^{i}_{n+1}(\{u_{n+1}\})\}=\{F^{a}_{n+1}\} \tag{17.30}$$

Note that the internal load $\{F^{i}_{n+1}(\{u_{n+1}\})\}$ is dependent on the current displacement $\{u_{n+1}\}$ at time t_{n+1} . The linearized form of the previous equation can be obtained by the Newton-Raphson method resulting in following equation:

$$\left[(a_0[M]+a_1[C])+[K^{T}_{n+1}(\{u^{k}_{n+1}\})]\right]\{\Delta u^{k}_{n+1}\}=\{R_{n+1}(\{u^{k}_{n+1}\})\} \tag{17.31}$$

where

$\{R_{n+1}(\{u^{k}_{n+1}\})\}$ = the residual vector

$\{\Delta u^{k}_{n+1}\}$ = the displacement increment

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$\left[K_{n+1}^{T}\left(\left\{u_{n+1}^{k}\right\}\right)\right]$ = the tangent stiffness matrix

and

$$a_0=\frac{1}{\alpha\Delta t^2} \tag{17.32}$$

$$a_1=\frac{\delta}{\alpha\Delta t} \tag{17.33}$$

The same logic is extended to the Backward Euler scheme when used in conjunction with the Nonlinear Elasticity model.

17.6. Thermoelasticity Model

The structural model allows you to perform thermoelasticity analyses, in order to simulate the effects of thermal load on solid structure deformation.

Note that thermal effects are only available for the linear or nonlinear structural model if the energy equation is enabled. Setting up the intrinsic FSI simulation is described in Modeling Fluid-Structure Interaction (FSI) Within Fluent in the *Fluent User's Guide* of the Fluent User's Guide.

For details about the theoretical basis of the thermoelasticity structural model, see the following sections:

17.6.1. Constitutive Equations

17.6.2. Finite Element Discretization

17.6.1. Constitutive Equations

The thermoelastic constitutive equations are:

$$\varepsilon_t=\varepsilon+\{\alpha\}\Delta T \tag{17.34}$$

$$\sigma_t=\sigma-\{\beta\}\Delta T \tag{17.35}$$

where:

ε_t = total strain vector

ΔT = temperature change = $T-T_{ref}$

T_{ref} = starting (reference) temperature

σ_t = total stress vector

$\{\alpha\}$ = vector of coefficients of thermal expansion

$\{\beta\}$ = vector of thermoelastic coefficients = $[D]\{\alpha\}$

$[D]$ = elastic stiffness matrix

17.6.2. Finite Element Discretization

In order to include thermal effects into the finite element matrix equation (Equation 17.7 (p. 830)), an additional term is added leading to the following formulation:

$$[K]\{u\}+[K^{uT}]\{\Delta T\}=\{f\} \tag{17.36}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

where:

$\{\Delta T\}$ = temperature vector

$[K^{uT}]$ = element thermoelastic stiffness matrix

The element thermoelastic stiffness matrix is defined as:

$$[K^{uT}] = -\int [B]^T \{\beta\} \{N\}^T dV \tag{17.37}$$

where the definition for the strain-displacement matrix $[B]$ and element shape functions $\{N\}$ follow the same rules as for the standard discretization theory.

17.7. Coupling of the Structural Model with the Battery Swelling Model

The structural model can be coupled with the battery swelling model to simulate battery cell deformation caused by electrochemical effects and deformation of the battery electrode layer. Battery swelling effects are only available for the linear or nonlinear structural models and when the battery swelling model is enabled as outlined in Inputs for the Newman's P2D Model in the *Fluent User's Guide*.

To simulate battery swelling effects, the structural model works in conjunction with the battery swelling model. The battery swelling model provides the electrode layer strain scalar ε_{swell} and the electrode layer normal vector $\underline{N}$ to the structural model to solve for the swelling strain tensor $\underline{\underline{\varepsilon}}_{swell}$. momentum balance, and swelling stress tensor $\underline{\underline{\sigma}}_{swell}$. After the structural model computes the above terms, they are provided to the battery swelling model to solve the relevant E-chem submodel equations. The variables that are passed between the structural model and the battery swelling model are referred to as coupling variables.

When using a dynamic mesh, the electrode layer normal vector $\underline{N}$ is transformed into the deformed frame battery layer normal vector $\underline{n}$ following the local rotation of the battery geometry and formulated as:

$$\underline{n} = \left(\underline{\underline{I}} + \frac{1}{2}\left(\underline{\underline{\nabla u}} - \underline{\underline{\nabla u}}^T\right)\right)\underline{N} \tag{17.38}$$

with $\underline{\underline{I}}$ being the identity matrix and $\underline{\underline{\nabla u}}$ being the deformation gradient. Note that when a dynamic mesh is not used for the battery geometry, $\underline{n}$ will be equal to $\underline{N}$.

The swelling strain tensor $\underline{\underline{\varepsilon}}_{swell}$ is then reconstructed from the electrode layer strain scalar ε_{swell} and $\underline{n}$ as:

$$\underline{\underline{\varepsilon}}_{swell} = \varepsilon_{swell}\, \underline{n} \otimes \underline{n} \tag{17.39}$$

The effective stress induced by battery swelling $\underline{\underline{\sigma}}_{swell}$ is then obtained from $\underline{\underline{\varepsilon}}_{swell}$ using the linear isotropic elasticity equation (Equation 17.3 (p. 829)). The following formulation is recommended for the Young's modulus in Equation 17.3 (p. 829) to ensure the coupling of variables is consistent with the battery swelling model:

$$E_{eff} = \frac{l_n + l_s + l_p}{\frac{l_n}{E_n} + \frac{l_s}{E_s} + \frac{l_p}{E_p}} \tag{17.40}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

where E is the Young's modulus and l is the thickness of the battery sandwich layer component, the subscripts $n.s.$ and p denote the battery anode, separator, and cathode, respectively (see Battery Swelling Model (p. 892). specifically Modeling Swelling in the E-Chem Standalone Mode (p. 897)), and E_{eff} denotes the effective Young's modulus used in the linear elasticity formulation.

To solve for the total stress $\underline{\underline{\sigma}}_t$ on the battery geometry, the momentum equation is solved with the swelling stress included and defined as:

$$0 = -\rho \underline{a} + \nabla \cdot \left(\underline{\underline{\sigma}} - \underline{\underline{\sigma}}_{swell} \right) \tag{17.41}$$

With the momentum equation solved, the total stress $\underline{\underline{\sigma}}_t$ on the battery geometry is defined as:

$$\underline{\underline{\sigma}}_t = \underline{\underline{\sigma}} - \underline{\underline{\sigma}}_{swell} \tag{17.42}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Chapter 18: Eulerian Wall Films

The Eulerian Wall Film (EWF) model can be used to predict the creation and flow of thin liquid films on the surface of walls. This chapter presents an overview of the theory and the governing equations for the methods used in the Eulerian Wall Film (EWF) model.

18.1. Introduction

18.2. Mass, Momentum, and Energy Conservation Equations for Wall Film

18.3. Passive Scalar Equation for Wall Film

18.4. Numerical Schemes and Solution Algorithm

For more information about using the Eulerian Wall Film model, see Modeling Eulerian Wall Films in the User's Guide.

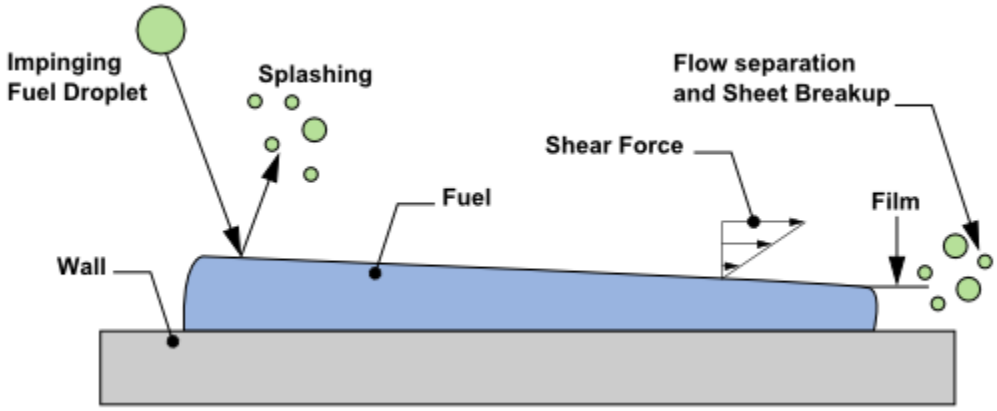
18.1. Introduction

The fundamental picture of a film model consists of a two dimensional thin film of liquid on a wall surface, as shown in Figure 18.1: Subgrid Processes That Require a Wall Film Model (p. 840). A thin liquid film forms as liquid drops impinge on a solid surface in the domain. There are several possible outcomes from this impingement: *stick*, where the droplet impacts the wall with little energy and remains nearly spherical; *rebound*, where the drop leaves the surface relatively intact but with changed velocity; *spread*, where the drop hits the wall with moderate energy and spreads out into the film; and *splash*, where part of the impinging drop joins the film and another part of the drop leaves the wall in several other smaller drops.

The thin film assumption is one usually made in both Eulerian and Lagrangian film modeling approaches, specifically that the thickness of the film is small compared to the radius of curvature of the surface so that the properties do not vary across the thickness of the film and that films formed are thin enough so that the liquid flow in the film can be considered parallel to the wall, with an assumed quadratic shape. Further modeling assumptions are needed to close the problem.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Figure 18.1: Subgrid Processes That Require a Wall Film Model



18.2. Mass, Momentum, and Energy Conservation Equations for Wall Film

Conservation of mass for a two dimensional film in a three dimensional domain is:

$$\frac{\partial \rho_l h}{\partial t} + \nabla_s \cdot \left(\rho_l h \vec{V}_l \right) = \dot{m}_s \tag{18.1}$$

where ρ_l is the liquid density, h the film height, ∇_s the surface gradient operator, $\vec{V}_l$ the mean film velocity and $\dot{m}_s$ the mass source per unit wall area due to droplet collection, film separation, film stripping, and phase change.

Conservation of film momentum is given,

$$\frac{\partial \rho_l h \vec{V}_l}{\partial t} + \nabla_s \cdot \left(\rho_l h \vec{V}_l \vec{V}_l + \overline{\overline{D}}_V \right) = -h \nabla_s P_L + \rho_l h \vec{g}_\tau + \frac{3}{2} \vec{\tau}_{fs} - \frac{3\mu_l}{h} \vec{V}_l + \dot{\vec{q}}_s + \vec{\tau}_{\theta_w} \tag{18.2}$$

where

$$P_L = P_{gas} + P_h + P_\sigma$$
$$P_h = -\rho h (\vec{n} \cdot \vec{g})$$
$$P_\sigma = -\sigma \nabla_s \cdot (\nabla_s h)$$

The terms on the left hand side of Equation 18.2 (p. 840) represent transient and convection effects, respectively, with tensor $\overline{\overline{D}}_V$ denoting the differential advection term computed on the basis of the quadratic film velocity profile representation [286] (p. 1069), [287] (p. 1069). On the right hand side, the first term includes the effects of gas-flow pressure, the gravity component normal to the wall surface (known as *spreading*), and surface tension; the second term represents the effect of gravity in the direction parallel to the film; the third and fourth terms represent the net viscous shear force on the gas-film and film-wall interfaces, based on the quadratic film velocity profile representation; the fifth term is associated with droplet collection or separation; and the last term is the surface force due to film liquid surface tension and contact angle. Note that in arriving at the shear and viscous terms on the RHS, a parabolic film velocity profile has been assumed.

Conservation of film energy is given as:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$\frac{\partial \rho_l h T_f}{\partial t} + \nabla_s \cdot \left(\rho_l h T_f \vec{V}_l + \vec{D}_T \right) = \frac{1}{C_p} \left[\frac{2k_f}{h} \left(T_s + T_w - 2T_m \right) + \dot{q}_{imp} + \dot{m}_{vap} L \right] \tag{18.3}$$

In the above equation, T_f is the average film temperature, and vector $\vec{D}_T$ is the differential advection term computed using the representation of quadratic film velocity and temperature profiles. On the right hand side, the first term inside the bracket represents the net heat flux on the gas-film and film-wall interfaces, with T_s and T_w as the film surface and wall temperatures, respectively, and T_m as the film half depth temperature, all computed from the film temperature profile representation and thermal boundary conditions at gas-film and film-wall interfaces. $\dot{q}_{imp}$ is the source term due to liquid impingement from the bulk flow to the wall. $\dot{m}_{vap}$ is the mass vaporization or condensation rate and L is the latent heat associated with the phase change.

Equation 18.1 (p. 840) and Equation 18.2 (p. 840) form the foundation of EWF modeling, with the solution of Equation 18.3 (p. 841) being optional only when thermal modeling is desired. These equations are solved on the surface of a wall boundary. Since the film considered here is thin, the lubrication approximation (parallel flow) is valid and therefore these equations are solved in local coordinates that are parallel to the surface.

Subsequent sections present the EWF sub-models and numerical solution procedures adopted in Ansys Fluent.

18.2.1. Film Sub-Models

The Eulerian wall film model can interact with the discrete phase model (DPM), VOF multiphase model, and Eulerian multiphase model through source terms to the film equations. During the interaction with the DPM model, the discrete particles are collected to form a wall film. The discrete particles can splash when interacting with a film boundary, creating additional particles in the same way as described in the Lagrangian film model. Additional particles can be created when the film separates from the wall, or when the shear stress is sufficient that large particles can be stripped from the film. Mass leaving the film surface by separation or stripping is accounted for through source terms to the film equations. In the Eulerian multiphase interaction, a secondary phase in the multiphase flow is captured on solid surfaces, forming liquid films. Collection efficiency can be computed for the solid surfaces. Mass and momentum leaving the liquid film due to separation and stripping is added to the secondary phase in the multiphase flow. In the interaction with the VOF multiphase model, film liquid excess of a specified limit can be modeled using the VOF model, and a secondary phase fluid in a VOF multiphase flow with small volumes below a specified limit can modeled using the wall film model.

18.2.1.1. DPM Collection

Discrete particle streams or discrete particles hitting a face on a wall boundary are absorbed into the film. When particles are absorbed, their mass and momentum are added to the source terms in Equation 18.1 (p. 840) and Equation 18.2 (p. 840), continuity and momentum equations, respectively. The mass source term is given by

$$\dot{m}_s = \dot{m}_p \tag{18.4}$$

where $\dot{m}_p$ is the flow rate of the particle stream impinging on the face. The momentum source term is

$$\dot{\vec{q}}_s = \dot{m}_p \cdot \left(\vec{V}_p - \vec{V}_l \right) \tag{18.5}$$

where $\vec{V}_p$ denotes the velocity of the particle stream and $\vec{V}_l$ denotes the film velocity.

18.2.1.2. Particle-Wall Interaction

Impingement of particles on the film wall boundary may result in particle rebounding, splashing or being absorbed to form wall film. Details of the particle interaction with the wall boundary is described in the Lagrangian film model (Interaction During Impact with a Boundary (p. 500)).

18.2.1.3. Film Separation

The film can separate from an edge if two criteria are met – first that the angle between faces is sufficiently large and second if the film inertia is above a critical value (defined by you). If separation occurs, a source term in the film equation is used to remove mass and momentum from the face corresponding to the edge upstream of where the separation occurs. This source term is accumulated at each film time step, and particle streams are created at DPM iterations using this source. New particle streams are created only during the DPM iteration, however the source terms due to film separating from an edge are updated each film step.

18.2.1.3.1. Separation Criteria

Based on the work by Foucart, separation can occur at an edge if a critical angle, θ_{critical}, is exceeded and a Weber number based on the film, We_f, is above a minimum value.

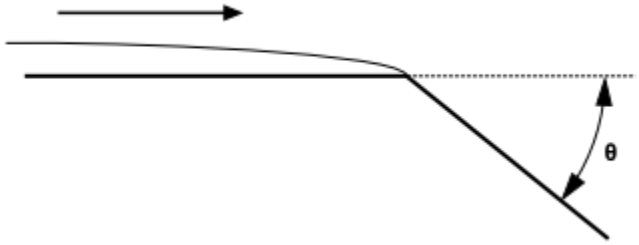
The Weber number is defined as,

$$We_f = \frac{\rho_l h \left|\vec{V}_l\right|^2}{\sigma} \tag{18.6}$$

and σ is surface tension of the film. The separation criteria become,

$$\begin{cases} \theta > \theta_{critical} \\ We_f > We_{critical} \end{cases} \tag{18.7}$$

Figure 18.2: Separation Criteria



Once separation occurs, you can specify three different models to calculate the number and diameter of the shed particle stream at an edge, based on work by Foucart ([188] (p. 1063)), O'Rourke ([492] (p. 1081)) and Friedrich ([193] (p. 1064)).

18.2.1.3.1.1. Foucart Separation

The Foucart model for edge separation assumes that the drop diameter is given by

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$d=\left(\frac{3h^2l}{4\pi}\right)^{\frac{1}{3}} \tag{18.8}$$

where l is the length of the edge and h the height of the film. The flow rate q of the particles from the film is equal to the mass flux of the film across the edge

$$q=V_l \bullet \hat{n}_e hl\rho_l \tag{18.9}$$

where $\hat{n}_e$ is the inward pointing normal from the edge centroid to the face center. The particles separated from an edge using Foucart's criterion are given the velocity of the film at the edge where the separation occurs.

18.2.1.3.1.2. O'Rourke Separation

The O'Rourke model for edge separation assumes that the drop diameter is equal to the height of the film at the edge, and the flow rate of the particles from the film is equal to the mass flux of the film across the edge.

18.2.1.3.1.3. Friedrich Separation

Friedrich's separation model assumes that the drop diameter is equal to the height of the film at the edge, but the flow rate of the particles from the film is calculated from an experimentally derived force ratio between the inertia of the film and the surface tension forces.

The force ratio is given by

$$F_{ratio}=\frac{We_f}{1+{}^{1}\!/_{\sin(\theta)}+Fr_f We_f\left({}^{L_b}\!/_{h}\right)\left({}^{1}\!/_{\tan(\theta)}\right)} \tag{18.10}$$

where $Fr_f=\frac{gh}{V_l^2}$ and the characteristic breakup length L_b is given by:

$$L_b=0.0388h^{0.5}\mathrm{Re}_h^{0.6}We_{rel}^{-0.5} \tag{18.11}$$

and the relative Weber number We_{rel} and the film flow Reynolds number are defined as follows,

$$We_{rel}=\frac{\rho_{gas}h(V_{gas}-V_l)^2}{2\sigma} \tag{18.12}$$

$$\mathrm{Re}_h=\frac{\rho_f V_l h}{\mu_l} \tag{18.13}$$

Unlike the O'Rourke or Foucart separation methods, the mass flux into the particle stream is an experimentally derived percentage of the flux across the edge as given in Equation 18.9 (p. 843). This percentage is assumed to be a function of the force ratio given by:

$$\gamma=1-\exp\left(0.411544\left(\frac{1-F_{ratio}^2}{F_{ratio}}\right)\right) \tag{18.14}$$

which is a curve fit to the data in Friedrich, et.al. ([193] (p. 1064)) and is valid for force ratios greater than one (that is, separated flows). If the force ratio in the Friedrich correlation is less than one, no separation takes place.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

18.2.1.4. Film Stripping

Film stripping occurs when high relative velocities exist between the gas phase and the liquid film on a wall surface. At sufficiently high shear rates, Kelvin-Helmholtz waves form on the surface of the film and grow, eventually stripping off droplets from the surface. The model for this behavior in the current implementation of the wall film model in Ansys Fluent is based on work by Lopez de Bertodano, et al. ([387] (p. 1075)) and Mayer ([414] (p. 1077)).

As described in Mayer, in a thin liquid sheet in a shear flow, waves of length λ are formed due a Kelvin-Helmholtz type of instability, but the waves are damped out by viscous forces in the film. The balance of wave growth and damping provides a term for the frequency ω

$$\omega = \frac{f}{\lambda^{1/2}} - \frac{\nu}{\lambda^2} \tag{18.15}$$

where

$$f = \sqrt{\frac{\pi}{2}} \frac{\beta \rho_g V_g^2}{\sqrt{\sigma \rho_l}} \tag{18.16}$$

with β as a sheltering parameter with a value of 0.3, and

$$\nu = \frac{8\pi^2 \mu_l}{\rho_l} \tag{18.17}$$

In Equation 18.16 (p. 844), V_g is the gas velocity.

For waves to grow and eventually break off from the film, ω must be greater than zero which implies that the minimum wave length for growing waves should be:

$$0 = \frac{f}{\lambda^{1/2}} - \frac{\nu}{\lambda^2} \Rightarrow \lambda_{\min} = \left(\frac{\nu}{f}\right)^{2/3} = 2\pi \sqrt[3]{16} \left(\frac{\mu_l \sqrt{\sigma/\rho_l}}{\beta \rho_g V_g^2}\right)^{2/3} \tag{18.18}$$

An expression for frequency from Taylor's theory can be derived using linear stability analysis, and is given by Lopez de Bertodano et al [387] (p. 1075)

$$\omega = 0.384 \frac{\rho_g V_g^3}{\sigma} \sqrt{\frac{\rho_g}{\rho_l}} \tag{18.19}$$

Following Mayer, the average droplet size of the stripped droplets is

$$\overline{d} = F\overline{\lambda} = F \cdot \frac{9}{2} \lambda_{\min} \tag{18.20}$$

where F (with a default value of 0.14) is a numerical factor that is equivalent to the **Diameter Coefficient** in the graphical user interface.

To obtain a mass flow rate at the surface of the liquid, it is assumed that $\dot{m} \cong \frac{1}{2} \rho_l \overline{\lambda} \omega$ where $\overline{\lambda}$ is the average wavelength of the disturbance. Mayer uses the expression $\overline{\lambda} = 4.5\lambda_{\min}$ which results in the following expression for the flow rate of drops stripped from the surface:

$$\dot{m} = C \frac{9}{2} \rho_l \omega \lambda_{\min} \tag{18.21}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

where C (with a default value of 0.5) is a numerical factor that is equivalent to the **Mass Coefficient** in the graphical user interface. Mass sources are accumulated during the course of the film calculation according to the above expression, and are injected into the free stream with the velocity of the film and the calculated mass flow rate from the face during the DPM iteration.

No mass will be taken from the film unless the critical shear stress (defined in the graphical user interface) is exceeded on the face where liquid film exists.

18.2.1.5. Secondary Phase Accretion

In an Eulerian multiphase flow, the secondary phase to be collected on a solid surface must have the same material as defined in the Eulerian wall film model. When the secondary phase is captured by the wall surface, its mass and momentum are removed from the multiphase flow, and added as source terms to the continuity and momentum equations, Equation 18.1 (p. 840) and Equation 18.2 (p. 840) respectively, of the wall film.

The mass source term is given by

$$\dot{m}_s = \alpha_d \rho_d V_{dn} A \tag{18.22}$$

where α_d is the secondary phase volume fraction, ρ_d is the secondary phase density, V_{dn} is the phase velocity normal to the wall surface, and A is the wall surface area.

The momentum source is give by

$$\vec{\dot{q}}_s = \dot{m}_s \vec{V}_d \tag{18.23}$$

where $\vec{V}_d$ is the secondary phase velocity vector.

The collection efficiency of the secondary phase is computed as

$$\beta = \frac{\rho_d V_{dn}}{C_d V_d} \tag{18.24}$$

where C_d and V_d are the reference (far field) secondary phase concentration and velocity, respectively.

18.2.1.6. Coupling of Wall Film with Mixture Species Transport

The Eulerian wall film model can also be coupled with the mixture species transport model to consider phase changes between film material (liquid) and the gas species (vapor). The phase change rate is modeled using following models:

- Diffusion Balance Model

  The rate of phase change is governed by

  $$\dot{m}_{phase} = \frac{(\rho D/\delta)}{\rho D/\delta + C_{phase}} C_{phase} (y_{sat} - y_i) \quad \left[kg/m^2\text{-}s\right] \tag{18.25}$$

  where ρ is the density of the gas mixture, D is the mass diffusivity of the vapor species, δ is the cell-center-to-wall distance, C_{phase} is the phase change constant, and y_i represents the cell-center mass fraction of the vapor species. The saturation species mass fraction is computed as

$$y_{sat} = \frac{P_{sat}(T)}{P}\frac{M_i}{\overline{M}} \tag{18.26}$$

where P is the absolute pressure of the gas mixture and, M_i and $\overline{M}$ is the molecular weight of the vapor species and the mixture, respectively. The saturation pressure P_{sat} is a function of temperature only. By default, water vapor is assumed to be the vapor species and its saturation pressure is computed, following Springer et al ([623] (p. 1089)), as

$$\begin{aligned}\log_{10} P_{sat} = &-2.1794+0.02953(T-273.17)-9.1837\text{x}10^{-5}(T-273.17)^2 \\ &+1.4454\text{x}10^{-7}(T-273.17)^3\end{aligned} \tag{18.27}$$

However, you can replace the above default formulation by using a user-defined property when defining the vapor material in the **Materials** task page.

It is clear from Equation 18.25 (p. 845) that when the vapor mass fraction exceeds the saturation mass fraction, condensation takes place (that is, negative $\dot{m}_{phase}$) instead, when the vapor mass fraction is less than the saturation mass fraction, the liquid film vaporizes. The phase change constant in Equation 18.25 (p. 845) takes different values for condensation and vaporization, respectively.

$$C_{phase} = \begin{cases} C_{con} \\ C_{vap} h \cdot 10^4 \end{cases} \tag{18.28}$$

where the film height h is used to prevent generating vapor without the presence of liquid.

- Wall Boundary Layer Model

  Details about the Wall Boundary Layer model are provided in Mass Transfer from the Film (p. 518).

- User Defined Model

  User-defined phase change rates can be specified through user-defined profile UDFs. Note that such UDFs must return the relevant rates as positive values in kg/s. The UDF returned rate will be converted into kg/s/m^2 with appropriate signs internally in the code.

18.2.1.7. Coupling of Eulerian Wall Film with the VOF Multiphase Model

When the liquid film accumulation becomes prominent, and the film thickness exceeds a specified maximum limit, the EWF model may not yield valid computational results. The ability to connect the EWF and VOF multiphase models enables the transition of the film liquid solution between the two models such that the solution of the accumulated film liquid can be carried out by the VOF multiphase model. Moreover, in VOF modeling, there are situations where the volume of fluid near a wall boundary can become very small. Resolving such small fluid volumes using the VOF model can be impractical and expensive. In such cases, a VOF to wall film transition allows the small volumes of fluid to be modeled using the EWF model.

Ansys Fluent provides the capability to couple the EWF and VOF multiphase models allowing a hybrid solution approach for such cases.

During an EWF and VOF model transition, mass, momentum and energy terms of the film and the secondary phase liquid are transferred between the two models. The transition can be based on one of the two transition bases:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

- A transition based on the volume fraction uses the film or secondary phase liquid volume fraction in the computation cells next to film walls
- A transition based on the height fraction uses the ratio of either the film or the equivalent film half height to the cell center distance to film walls.

With both transition bases, a weighted film height threshold is computed from the specified transition criterion and used to determine the onset of a transition.

18.2.2. Partial Wetting Effect

The effect of partial wetting by film liquid describes the behaviour such as rivulets, dry patches, and film rupture on a wall surface and is modeled using the partial wetting model described by Meredith [432] (p. 1078).

The surface force due to film surface tension $\vec{\tau}_{\theta_w}$ in Equation 18.2 (p. 840) is given as:

$$\vec{\tau}_{\theta_w}=\beta\sigma\left(1-cos\theta_w\right)\nabla_s w \tag{18.29}$$

where σ is the film surface tension, β is an empirical parameter to account for discrepancies between the model and observed film behaviour, and w represents the film wet area fraction.

The film contact angle θ_w is sampled randomly from a Gaussian distribution over the film wall characterized by the mean contact angle θ_m and its standard deviation $\alpha\theta_m$ (where α is a percentage fraction factor). The contact angle can also be specified using user-defined functions (UDFs).

The film wet area fraction is given as:

$$w=\begin{cases}1, & h>h_c\\ 0, & h\le h_c\end{cases} \tag{18.30}$$

where h_c is the critical film thickness. If the thickness of the liquid film is above the critical value h_c, the wall is considered to be wet. If the film thickness is below this value, the wall is considered to be dry. In the Ansys Fluent Eulerian Wall Film (EWF) model, h_c takes the value of 1e-10m.

18.2.3. Boundary Conditions

Boundary conditions for Equation 18.1 (p. 840) and Equation 18.2 (p. 840), film mass and momentum equations, are specified only at the wall on which film modeling is enabled. There are two types of conditions: one is the fixed mass and momentum fluxes and the other specified initial film height and velocity. When the fluxes are specified, they are added to the equations as source terms. When the initial film height and velocity are specified, these values are enforced only at the first film time step.

18.2.4. Obtaining Film Velocity Without Solving the Momentum Equations

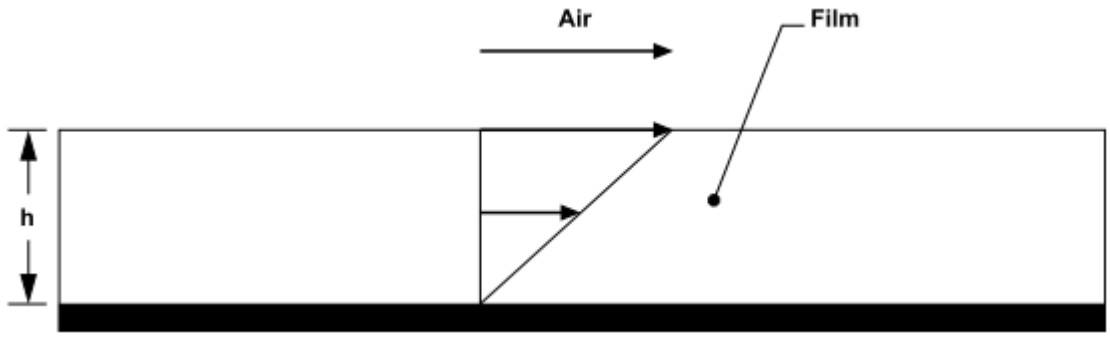
Equation 18.2 (p. 840) is used to obtain film average velocity through momentum conservation. However, one can estimate the film velocity without solving Equation 18.2 (p. 840). In this case, the film velocity is assumed to have a component that is driven by the shear due to the external flow and a component that is driven by gravity.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

18.2.4.1. Shear-Driven Film Velocity

As shown in Figure 18.3: Shear-Driven Film Velocity (p. 848), it is assumed that the film velocity varies linearly from the wall to the air-film interface.

Figure 18.3: Shear-Driven Film Velocity



By equating the two shear forces at the film-air interface, one from the air side and one from the film side, we have,

$$\vec{\tau}_{air}=\mu_l\frac{\partial\vec{V}}{\partial n}=\mu_l\frac{\vec{V}_s}{h} \tag{18.31}$$

where $\vec{V}_s$ is the surface velocity. The film average velocity is therefore,

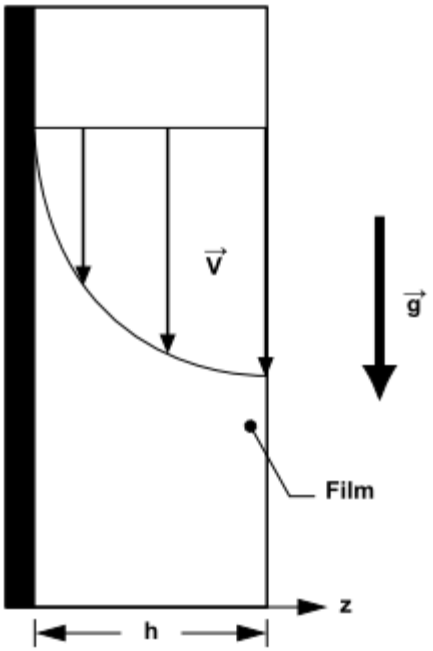
$$\vec{V}_l=\frac{\vec{\tau}_{air}h}{2\mu_l} \tag{18.32}$$

18.2.4.2. Gravity-Driven Film Velocity

It is assumed that the film movement is entirely controlled by the balance of gravity force and the wall shear force, and that the film velocity has a parabolic profile with a zero-gradient at the air-film interface, as shown in Figure 18.4: Gravity-Driven Film Velocity (p. 849).

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Figure 18.4: Gravity-Driven Film Velocity



Assuming no-slip at the wall, the parabolic film velocity becomes $\vec{V}(z)=\vec{a}z^2+\vec{b}z$.

Balancing the weight of the film and the wall shear force, we have,

$$\rho_l Ah\vec{g}=A\mu_l\frac{\partial\vec{V}}{\partial z}\bigg|_{z=0}=\mu_l(2\vec{a}z+\vec{b})_{z=0} \tag{18.33}$$

therefore,

$$\vec{b}=\frac{\rho_l h}{\mu_l}\vec{g} \tag{18.34}$$

Applying the zero-gradient assumption at the air-film interface, we have,

$$\vec{a}=-\frac{\vec{b}}{2h}=-\frac{\rho_l}{2\mu_l}\vec{g} \tag{18.35}$$

Thus the gravity-induced film velocity is,

$$\vec{V}(z)=\frac{\rho_l h}{\mu_l}z(1-\frac{z}{2h})\vec{g} \tag{18.36}$$

The film average velocity is,

$$\vec{V}_l=\frac{\rho_l h^2}{3\mu_l}\vec{g} \tag{18.37}$$

18.3. Passive Scalar Equation for Wall Film

Transport equation for a passive scalar in wall film is given by:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$\frac{\partial \rho_l h Y}{\partial t} + \nabla_s \cdot \left(\rho_l h \vec{V}_l Y \right) = \nabla_s \cdot \left[\Gamma \nabla_s \left(hY \right) \right] + \dot{m}_s \tag{18.38}$$

where,

Y = arbitrary passive scalar

h = film height

$\vec{V}_l$ = mean film velocity

ρ_l = liquid density

m_s = scalar related mass source term per unit area

Γ = diffusivity for the scalar

If you select the **Passive Scalar** option in the **Eulerian Wall Film** dialog box, the Fluent solver uses Equation 18.38 (p. 850) to solve the transportation of a passive scalar due to film convection and scalar diffusion. Note that the solution of the passive scalar transport has no influence on the solution of wall film equations. For more details, see Setting Eulerian Wall Film Model Options in the *Fluent User's Guide*.

18.4. Numerical Schemes and Solution Algorithm

This section describes the temporal and spatial differencing schemes, as well as the solution algorithms used with the Eulerian Wall Film (EWF) model.

18.4.1. Temporal Differencing Schemes
18.4.2. Spatial Differencing Schemes
18.4.3. Solution Algorithm
18.4.4. Coupled Solution Approach

18.4.1. Temporal Differencing Schemes

The following temporal differencing schemes are available for the Eulerian Wall Film (EWF) model.

18.4.1.1. First-Order Explicit Method
18.4.1.2. First-Order Implicit Method
18.4.1.3. Second-Order Implicit Method

18.4.1.1. First-Order Explicit Method

Defining *F* and *G* as follows, with the subscript i indicating the previous time-step values:

$$\begin{aligned} F_i &\equiv \left[-\nabla_s \cdot \left(\rho_l h \vec{V}_l \right) + \dot{m}_s \right]_i \\ G_i &\equiv \left[-h \nabla_s P_L + \rho_l h \vec{g}_\tau + \frac{3}{2} \vec{\tau}_{fs} - \frac{3\mu_l}{h} \vec{V}_l + \dot{\vec{q}}_s + \vec{\tau}_{\theta_w} - \nabla_s \cdot \left(\rho_l h \vec{V}_l \vec{V}_l + \bar{\bar{D}}_V \right) \right]_i \end{aligned} \tag{18.39}$$

we have the following discretized film mass and momentum equations, with subscript $i+1$ representing the current time-step values, and Δt the time step being used for film computations.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$\begin{aligned}\frac{(\rho h)_{i+1}-(\rho h)_i}{\Delta t} &= F_i \\ \frac{\left(\rho_l h\vec{V}_l\right)_{i+1}-\left(\rho_l h\vec{V}_l\right)_i}{\Delta t} &= -\left(\frac{3\mu_l}{h}\vec{V}_l\right)_{i+1}+G_i\end{aligned} \tag{18.40}$$

Film height and velocity are then computed as follows,

$$\begin{aligned}h_{i+1} &= \frac{\left(\rho_l h\right)_i+\Delta t F_i}{\left(\rho_l\right)_{i+1}} \\ \left(\vec{V}_l\right)_{i+1} &= \frac{\left(\rho_l h\vec{V}_l\right)_i+\Delta t G_i}{\left(\rho_l h\right)_{i+1}+\Delta t\frac{3\mu_l}{h_{i+1}}}\end{aligned} \tag{18.41}$$

The above set of equations complete the *explicit* differencing scheme in which film height is computed first, based upon values of F_i evaluated at the previous film time step; then film velocity is calculated using the latest film height and values of G_i evaluated at the previous film time step.

18.4.1.2. First-Order Implicit Method

With the explicit method, evaluations of F and G are done based upon the previous time step film height and velocity vector. In order to improve accuracy of this explicit method, a first-order implicit method is introduced in which F and G values are updated during an iterative loop within a film time step. This new method can be described as a predictor-corrector procedure. At the beginning, that is, the predictor step, the explicit scheme is used to compute film height and velocity vector,

Predictor:

$$\begin{aligned}h_{i+1}^0 &= \frac{\left(\rho_l h\right)_i+\Delta t F_i}{\rho_{li+1}^0} \\ \left(\vec{V}_l\right)_{i+1}^0 &= \frac{\left(\rho_l h\vec{V}_l\right)_i+\Delta t G_i}{\left(\rho_l h\right)_{i+1}^0+\Delta t\frac{3\mu_l}{h_{i+1}^0}}\end{aligned} \tag{18.42}$$

The superscript 0 indicates the first step in the iteration loop.

Corrector:

The latest film height and velocity vector are used to update F and G; then film height and velocity are recomputed,

$$\begin{aligned}h_{i+1}^{n+1} &= \frac{\left(\rho_l h\right)_i+\Delta t F\left[h_{i+1}^n,\left(\vec{V}_l\right)_{i+1}^n\right]}{\rho_{li+1}^{n+1}} \\ \left(\vec{V}_l\right)_{i+1}^{n+1} &= \frac{\left(\rho_l h\vec{V}_l\right)_i+\Delta t G\left[h_{i+1}^n,\left(\vec{V}_l\right)_{i+1}^n\right]}{\left(\rho_l h\right)_{i+1}^{n+1}+\Delta t\left(3\mu_l/h_{i+1}^{n+1}\right)}\end{aligned} \tag{18.43}$$

The superscript n+1 and n represent the current and the previous iterations, respectively. The iteration procedure ends with the following convergence criteria:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$\left|h_{i+1}^{n+1}-h_{i+1}^{n}\right|\le\varepsilon$$
$$\left|(u_\alpha)_{i+1}^{n+1}-(u_\alpha)_{i+1}^{n}\right|\le\varepsilon \tag{18.44}$$

where u_α represent each component of the velocity vector.

18.4.1.3. Second-Order Implicit Method

With the above discussed explicit and first-order implicit methods, time differencing is only first-order accurate. A second-order implicit method is introduced below. The iterative procedure is very similar to that used in the first-order implicit method, but two time-steps (i and i–1) are used for time differencing in the predictor step.

Predictor:

$$h_{i+1}^{0} = \frac{\left(\rho_l h\right)_i+(\Delta t/2)(3F_i-F_{i-1})}{\rho_{li+1}^{0}}$$
$$\left(\vec{V}_l\right)_{i+1}^{0} = \frac{\left(\rho_l h\vec{V}_l\right)_i+(\Delta t/2)(3G_i-G_{i-1})}{\left(\rho_l h\right)_{i+1}^{0}+\Delta t\left(3\mu_l/h_{i+1}^{0}\right)} \tag{18.45}$$

Corrector:

$$h_{i+1}^{n+1} = \frac{\left(\rho_l h\right)_i+(\Delta t/2)\left\{F\left[h_{i+1}^{n}(\vec{V}_l)_{i+1}^{n}\right]+F\left[h_{i+1}^{n-1}(\vec{V}_l)_{i+1}^{n-1}\right]\right\}}{\rho_{li+1}^{n+1}}$$
$$\left(\vec{V}_l\right)_{i+1}^{n+1} = \frac{\left(\rho_l h\vec{V}_l\right)_i+(\Delta t/2)\left\{G\left[h_{i+1}^{n}(\vec{V}_l)_{i+1}^{n}\right]+G\left[h_{i+1}^{n-1}(\vec{V}_l)_{i+1}^{n-1}\right]\right\}}{\left(\rho_l h\right)_{i+1}^{n+1}+\Delta t\left(3\mu_l/h_{i+1}^{n+1}\right)} \tag{18.46}$$

The iteration procedure ends with the following convergence criteria:

$$\left|h_{i+1}^{n+1}-h_{i+1}^{n}\right|\le\varepsilon$$
$$\left|(u_\alpha)_{i+1}^{n+1}-(u_\alpha)_{i+1}^{n}\right|\le\varepsilon \tag{18.47}$$

where u_α represent each component of the velocity vector.

The discretized expressions for film energy and passive scalar equations can be derived in a similar way as above.

18.4.2. Spatial Differencing Schemes

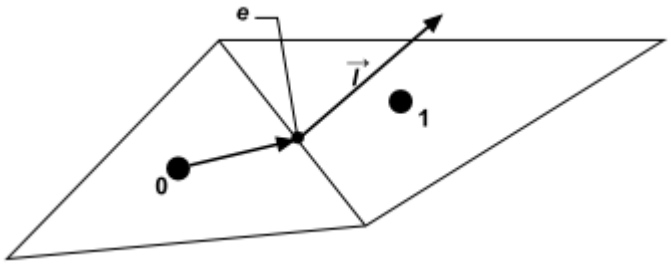
Computing F and G, defined in Equation 18.39 (p. 850), involves evaluations of spatial gradients for various quantities at face centers. Applying Green-Gauss theorem the gradient of a scalar φ at face center 0 is computed as,

$$\nabla\varphi=\frac{1}{A}\sum_e\varphi_e\vec{l}_e \tag{18.48}$$

where the subscript e represents the center of each edge of a film face, A the area of the face and $\vec{l}$ the length vector (whose magnitude is the length of the edge and whose direction is normal to the edge), as in the sketch below,

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Figure 18.5: Spatial Gradient



Clearly, at the center of the gradient calculation is how to obtain φ_e. The following describes the steps taken in the current implementation.

(1) Primary gradient computation

A primary gradient is calculated using Green-Gauss theorem by simply estimating edge center value as,

$$\varphi_e=\frac{1}{2}(\varphi_P+\varphi_E) \tag{18.49}$$

$$\nabla\varphi_P=\frac{1}{A}\sum_e \varphi_e\vec{l}_e \tag{18.50}$$

(2) Reconstruction of edge-center values

The edge-center values are then computed using either the first-order upwind or the second-order upwind schemes,

1st-order upwind: $\varphi_e^*=\varphi_{UP}$

2nd-order upwind: $\varphi_e^*=\varphi_{UP}+\vec{d}r_{UP}\cdot\nabla\varphi_P$

φ_{UP} is the value of φ at the UPWIND face center; $\vec{d}r_{UP}$ is the distance vector from the UPWIND face center to the edge center.

(3) Final gradient calculation

Finally the gradient at face center is computed as,

$$\nabla\varphi=\frac{1}{A}\sum_e \varphi_e^*\vec{l}_e \tag{18.51}$$

18.4.3. Solution Algorithm

While film movement is always tracked with time, the EWF model can be run with both steady and transient flow simulations.

18.4.3.1. Steady Flow

It is assumed that while flow affects its movement, the thin film does not, however, affect the flow field. In this case the flow field has converged and is kept unchanged (frozen flow field). The elapsed time for film development is increased by the film time step, Δt. This time step can either be a

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

user-specified constant or can be computed automatically using an adaptive time-stepping method with user-specified increase and decrease factors, depending upon the maximum Courant number, described as follows:

- When the computed maximum Courant number exceeds the user-specified value, the film time step is reduced by a decrease factor F_{dec} (default: 2).
- When the computed maximum Courant number is less than half of the user-specified value, the film time step is increased by an increase factor F_{inc} (default: 1.5).

18.4.3.2. Transient Flow

Within each flow time step, Δt_{flow}, a number of film sub-time steps are used to advance the film time to the same physical time of flow. The film sub-time step is determined as follows,

$$\Delta t = \frac{\Delta t_{flow}}{N_{film}} \tag{18.52}$$

N_{film} is the number of film time steps; Δt_{flow} is the flow time step.

Adaptive time stepping is achieved by adjusting the number of film time steps N_{film} using user-specified increase and decrease factors F_{inc} and F_{dec}:

- When the computed maximum Courant number exceeds the user-specified value, the number of film time steps N_{film} is increased by a factor of F_{dec}, hence reducing the film time step by factor F_{dec}.
- When the computed maximum Courant number is less than half of the user-specified value, the number of film time steps N_{film} is reduced by a factor of F_{inc}, hence increasing the film time step by factor F_{inc}.

  Note that the minimum number of film time steps that can be reduced to is 2.

Since film computation starts at the end of each flow time step, the effect of the film on flow is lagged by one flow time step.

18.4.4. Coupled Solution Approach

By default, the film mass and momentum equations are solved sequentially in the Eulerian Wall Film model. In situations where the film height and velocities are closely coupled, a segregated solution approach will not be able to resolve the specific features of the film, such as wavy surface of a free-falling film.

In the coupled solution approach, the film mass and momentum equations are solved in parallel, where film height and velocity solutions of the previous time step are used in the solution of both equations in the next time step.

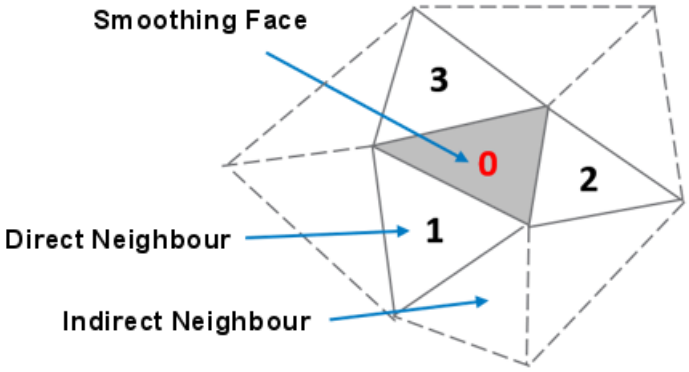
To improve the solution stability, smoothing is applied to variables used in film curvature and surface tension force calculations in the coupled solution approach. Variable smoothing is achieved by blending variable values from neighbouring faces in an iterative operation:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$\varphi_s = \frac{1}{w} \sum_{i=1}^{n} \left(\varphi_0 \left(1 - \alpha \right) + \varphi_i \alpha \right) \tag{18.53}$$

where φ_0 and φ_s refer to original and smoothed values of the film variable φ on a given face, respectively; φ_i refers to a value of φ on the i^{th} neighbouring face (direct or indirect), as illustrated in Figure 18.6: Face Value Smoothing (p. 855); α is a smoothing factor ranging from 0 to 1 (default = 0.5); and w is a weighting factor determined by the smoothing algorithm.

Figure 18.6: Face Value Smoothing



The weighted averaging is repeated as many times as specified by the **Smoothing Level** input parameter ranging from 1 to 4 (default = 2).

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information
of ANSYS, Inc. and its subsidiaries and affiliates.

Chapter 19: Electric Potential and Electrochemistry Models

This chapter outlines the basis of the implementation of the electric potential and electrochemistry models in Ansys Fluent. For information about using these models, see Modeling Electric Potential Field and Electrochemistry Models in the *Fluent User's Guide*.

The information in this chapter is organized into the following sections:

19.1. Electric Potential

19.2. Lithium-ion Battery Model

19.3. Electrolysis and H2 Pump Model

19.1. Electric Potential

19.1.1. Overview

Ansys Fluent can model problems involving the electric potential field by solving the electric potential equation, which can be solved in both fluid and solid zones. The electric potential solver can be used alone or in conjunction with other Ansys Fluent models. One example where the electric potential model could be coupled with other Ansys Fluent models is a discrete phase simulation of an electrostatic precipitator. To compute the electrostatic force of the charged particles, the electric potential equation must be solved.

The electric potential solver is automatically used with the built-in electrochemical reaction model allowing for the simulation of chemical and electrochemical reactions. For details on how the electrochemical reaction model is implemented in Ansys Fluent, refer to Electrochemical Reactions (p. 267). The electric potential solver is also used in the electrochemistry models (lithium-ion battery and electrolysis and H2 pump models).

If you want to use the electric potential solver together with other Ansys Fluent CFD solvers, you need to manually enable the potential equation as described in Setting Up the Electric Potential Model in the *Fluent User's Guide*.

The electric potential modeling capabilities allow Ansys Fluent to simulate a wide variety of phenomena associated with electric potential fields, such as electro-plating, corrosion, flow battery, and others. The physical equation used in this model is described in Electric Potential Equation (p. 858). Instructions for setting up and solving an electric potential field problem can be found in Setting Up the Electric Potential Model in the *Fluent User's Guide*. Details about using the electric potential solver within the electrochemical reaction model are provided in Electrochemical Reactions in the *Fluent User's Guide*.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

19.1.2. Electric Potential Equation

When the electric potential solver is enabled, Ansys Fluent solves the following electric potential equation:

$$\nabla \cdot \left(\sigma \nabla \varphi \right) + S = 0 \tag{19.1}$$

where

φ = electric potential

σ = electric conductivity in a solid zone or ionic conductivity in a fluid zone

S = source term

Note that Equation 19.1 (p. 858) does not have a transient term. To solve Equation 19.1 (p. 858), either electric potential or electric current needs to be specified at all external boundaries as described in Setting Up the Electric Potential Model in the *Fluent User's Guide*.

19.1.3. Energy Equation Source Term

When electric current flows through a medium, it produces heat. This phenomenon is called Joule heating. Joule heating generated inside the medium, S_{h1}, can be computed as:

$$S_{h1} = \frac{1}{\sigma} \left| \nabla \varphi \right|^2 \tag{19.2}$$

When contact resistance is defined on a wall, an extra Joule heating source term, S_{h2}, is added to the wall's neighboring cell:

$$S_{h2} = \frac{I^2 R f}{V} \tag{19.3}$$

where

I = current flux through the wall (A/m^2)

R = contact resistance (Ohm.m^2)

f = wall area

V = cell volume

When solving the potential equation (Equation 19.1 (p. 858)), you can add the Joule heating to the energy equation (Equation 5.1 (p. 172)).

19.2. Lithium-ion Battery Model

19.2.1. Overview

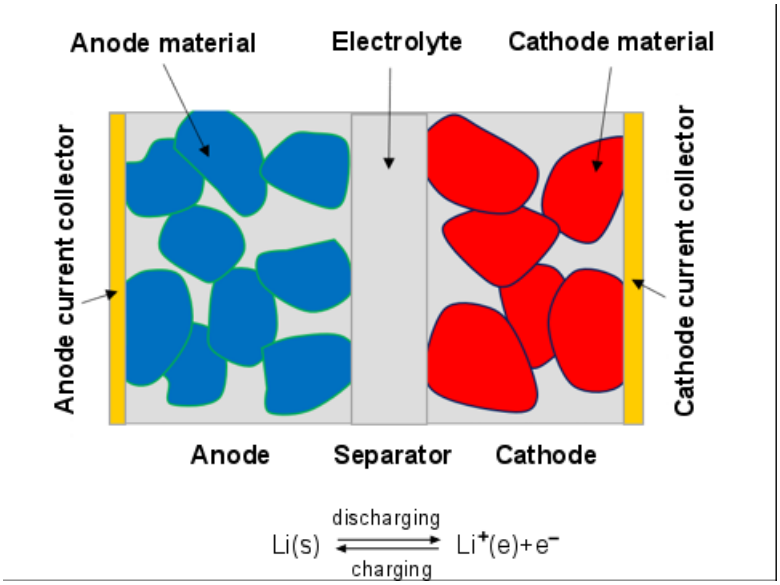
The lithium-ion battery model is designed to simulate detailed physics occurring at very small electrode scale. If you are interested in the thermal aspect of whole battery cell or a battery pack, consider using another modeling strategies such as dual potential Multi-Scale Multi-Domain (MSMD) battery model. For details about this model, see Battery Model (p. 871).

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

19.2.2. Lithium-ion Battery Model Theory

Lithium-ion battery is typically made by stacking/rolling-up many layers of the electrodes to form a sandwich-like structure. Each electrode consists of the anode current collector, anode, separator, cathode, and cathode current collector as shown in Figure 19.1: Schematic structure of an electrode pair (p. 859).

Figure 19.1: Schematic structure of an electrode pair



During discharge, lithium stored in the anode material diffuses out, undergoes electrochemical reaction at the solid-electrolyte interface (SEI), and dissolves in the electrolyte phase as lithium-ion species. Then lithium-ion species diffuses further from the anode side to the cathode side. As the lithium-ion species undergoes electrochemical reaction at the cathode SEI, lithium is intercalated into the cathode material. During the charge process, the lithium and lithium-ion move in the opposite direction.

The detailed lithium-ion battery model simulates the detailed physics occurring in the charging and discharging processes from first principles. The model is derived from the concentrated solution theory [471] (p. 1080), [212] (p. 1065). The governing equations are the mass and charge balance equations expressed as:

- Mass balance

$$\frac{\partial c}{\partial t} = \nabla \cdot \vec{N} \tag{19.4}$$

- Charge balance:

$$\nabla \cdot \vec{j} = 0 \tag{19.5}$$

where

c = either lithium concentration in the electrode or lithium-ion concentration in the electrolyte

$\vec{N}$ = mass flux vector

$\vec{j}$ = electric current vector

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

The mass and electric current flux can be computed as:

- In electrode zone:

$$\begin{aligned} \vec{j}_s &= -\sigma \nabla \phi_s \\ \vec{N}_s &= -D_s \nabla c_s \end{aligned} \tag{19.6}$$

- In electrolyte zone:

$$\begin{aligned} \vec{j}_e &= -\kappa \nabla \phi_e + \kappa_D \nabla \ln c_e \\ \vec{N}_{\mathrm{e}} &= -D_e \nabla c_e + \frac{\vec{j}_e t^+}{F} \end{aligned} \tag{19.7}$$

where

$\vec{j}_s$, $\vec{j}_e$ = electric current flux vectors in electrode solid zone and electrolyte fluid zone, respectively (A/m^2 s^{-1})

$\vec{N}_{\mathrm{s}}$ = lithium species mass flux vector in electrode solid zone (Kmol/m^2 s^{-1})

$\vec{N}_{\mathrm{e}}$ = lithium-ion species mass flux vector in electrolyte zone (Kmol/m^2 s^{-1})

ϕ_s, ϕ_e = electric potential in solid electrode and electrolyte, respectively (V)

c_s, c_e = lithium concentration in solid electrode and electrolyte, respectively (Kmol/m^3)

σ = electric conductivity in electrode solid (S m^{-1})

κ = ionic electric conductivity in electrolyte (S m^{-1})

D_s = mass diffusivity of lithium in electrode (m^2 s^{-1})

D_e = mass diffusivity of lithium-ion species in electrolyte (m^2 s^{-1})

F = Faraday constant (C/Kmol)

t^+ = lithium ion transference number

κ_D = diffusional conductivity

κ_D is defined by:

$$k_D = \frac{2RTk}{F}\left(1 - t^+\right)\left(1 + \frac{\partial\left(\ln f_{\pm}\right)}{\partial\left(\ln c\right)}\right) \tag{19.8}$$

where

R = Universal gas constant (J K^{-1} Kmol^{-1})

T = temperature (K)

$f_{\pm}$ = activity coefficient

The electrochemical reaction occurs at the SEIs. The reaction rate is determined by the Butler-Volmer equation:

$$i_{se} = i_0 F c_e^{\alpha_1}\left(c_{s,max} - c_s\right)^{\alpha_2} c_s^{\alpha_3}\left[\exp\left(\frac{\alpha_a F}{RT}\left(\phi_s - \phi_e - U\right)\right) - \exp\left(-\frac{\alpha_c F}{RT}\left(\phi_s - \phi_e - U\right)\right)\right] \tag{19.9}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

A linearized form can also be used:

$$i_{se}=i_0 F c_e^{\alpha_1}\cdot\left(c_{s,\,max}-c_s\right)^{\alpha_2} c_s^{\alpha_3}\left[\frac{(\alpha_{a+}\alpha_c)F}{RT}\left(\phi_s-\phi_e-\ U\right)\right] \tag{19.10}$$

where

i_0 = exchange current constant (consistent units)

α_a, α_c= anodic and cathodic transfer coefficients, respectively

α_1, α_2, α_3 = reaction rate exponents

U = equilibrium potential (V)

At the SEIs, although potential and lithium are not continuous, the mass and electric current fluxes must be continuous satisfying the following conditions:

$$\vec{j}_s\cdot\vec{n}=\vec{j}_e\cdot\vec{n}=i_{se} \tag{19.11}$$

$$\vec{N}_s\cdot\vec{n}=\vec{N}_s\cdot\vec{n}=\frac{i_{se}}{F} \tag{19.12}$$

Substituting the flux equations (Equation 19.6 (p. 860) and Equation 19.7 (p. 860)) into the mass and current conservation equations (Equation 19.4 (p. 859) and Equation 19.4 (p. 859)) yields the following:

- Mass conservation law:
  - Electrode:

$$\frac{\partial c_s}{\partial t}=\nabla\cdot\left(D_s\nabla c_s\right) \tag{19.13}$$

  - Electrolyte:

$$\frac{\partial c_e}{\partial t}=\nabla\cdot\left(D_e\nabla c_e\right)-\nabla\cdot\left(\frac{t^+\vec{j}_e}{F}\right) \tag{19.14}$$

- Charge conservation law:
  - Electrode:

$$\nabla\cdot\left(\sigma\nabla\phi_s\right)=0 \tag{19.15}$$

  - Electrolyte:

$$\nabla\cdot\left(\kappa\nabla\phi_e\right)-\nabla\cdot\left(\kappa_D\,\nabla\ln c_e\right)=0 \tag{19.16}$$

Ansys Fluent solves Equation 19.13 (p. 861) through Equation 19.16 (p. 861) together with the SEI conditions (Equation 19.11 (p. 861) and Equation 19.12 (p. 861) to obtain the electric and species fields.

For information about using the lithium-ion battery model, see Simulating the Lithium-ion Battery in the *Fluent User's Guide*.

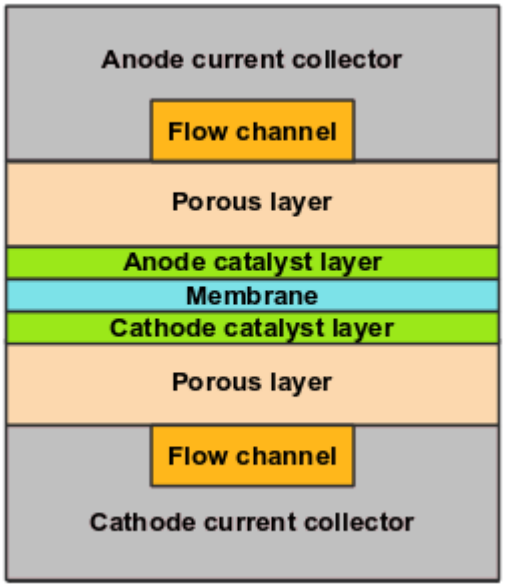
Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

19.3. Electrolysis and H2 Pump Model

19.3.1. Overview

The electrolysis and hydrogen (H2) pump model allows you to model polymer electrolyte membrane (PEM), hydrogen pump, and alkaline electrolysis devices [346] (p. 1073). Both PEM electrolysis and alkaline electrolysis produce gaseous hydrogen and oxygen from liquid water, while the hydrogen pump is used to purify and compress hydrogen through an electrochemistry process. With the electrolysis and H2 pump model, both active catalyst layers and ionic conducting electrolyte (also known as the membrane in electrolysis terminology) are included in the computational domain. To determine the physical domains that are included in the electrolysis and H2 pump model, a schematic of typically electrolysis device is shown in Figure 19.2: Schematic of an Electrolysis Device (p. 862).

Figure 19.2: Schematic of an Electrolysis Device



In the electrolysis and H2 pump model in Ansys Fluent, two electric potential fields are solved in the following zones:

- in the electrolyte zone and the catalyst layers
- in the catalyst layers, the porous electrode, and the current collectors

The flow field is only solved in the flow channels, porous electrodes, and catalyst layers. In the membrane, the flow field is not resolved since the membrane is used to separate the anode and cathode sides.

The rates of electrochemical reactions are computed in the catalyst layers at both the anode and the cathode. The current density value is estimated based on the cell voltage, which can be either directly specified or computed from a specified average current density.

The electrolysis devices differ in electrochemical reactions in catalyst layers:

- For PEM electrolysis, the electrochemical reactions are:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$Anode: \quad 2H_2O \rightarrow O_2 + 4H^+ + 4e^-$$
$$Cathode: \quad 4H^+ + 4e^- \rightarrow 2H_2$$

- For alkaline electrolysis, the electrochemical reactions are:

$$Anode: \quad 2OH^- \rightarrow 1/2\, O_2 + H_2O + 2e^-$$
$$Cathode: \quad 2H_2O + 2e^- \rightarrow H_2 + 2OH^-$$

- For the hydrogen pump, the electrochemical reactions are:

$$Anode: \quad H_2 \rightarrow 2H^+ + 2e^-$$
$$Cathode: \quad 2H^+ + 2e^- \rightarrow H_2$$

Although electrochemistry reactions often involve ionic species, the electrolysis and H2 pump model in Ansys Fluent does not track them to simplify modeling. That is, only the transport equations of neutral species are solved.

In Ansys Fluent, the following approaches are used for modeling the membrane electrode assemblies (MEA) zones (the catalyst layers and the membrane):

- Resolved

  In this approach, the MEA zones are included in the computational domain, and the dual potential equations are solved. For details, see Resolved Modeling Approach (p. 863).

- Unresolved 0D

  In this approach, the MEA zones are not included in the computational domain. Instead, they are modeled as a pair of wall and wall-shadow faces called “electrolyte interfaces". The species sources, energy sources and the sinks due to the electrochemical reactions are added to the adjacent computational cells. Because only one potential equation is solved, the computational cost is reduced. For details, see Unresolved 0D Modeling Approach (p. 868)

Note:

The resolved approach can be used only for unit electrolysis cell modeling, while the unresolved 0D approach can be used for modeling both unit electrolysis cells and electrolysis stacks.

19.3.2. Resolved Modeling Approach

This section presents the theory behind the resolved approach to modeling electrolysis and H2 pump devices. The following topics are discussed:

19.3.2.1. Electrochemistry Modeling

19.3.2.2. Multiphase Modeling

19.3.2.3. Heat Source

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

19.3.2.1. Electrochemistry Modeling

Electrochemistry modeling in electrolysis devices or a hydrogen pump is very similar to that in the fuel cell add-on module described in Electrochemistry Modeling in the *Fluent Theory Guide* (p. 901). The major difference is that in the fuel cell model, the electrical potential is fixed at zero on the anode side, while in the electrolysis and H2 pump model, it is fixed at zero on the cathode side.

In the Butler-Volmer equations (Equation 21.5 (p. 903) and Equation 21.6 (p. 903)), the concentration dependence of species is modeled in different electrolysis devices as follows:

- PEM electrolysis: the concentration dependence of liquid water is considered only on the anode side and is not modeled on the cathode side (γ_{cat}=0).
- Alkaline electrolysis: the concentration dependence of liquid water is considered only on the cathode side and is not modeled on the anode side (γ_{an}=0).
- H2 pump: the concentration dependence of hydrogen is considered only on the anode side and is not modeled on the cathode side (γ_{cat}=0).

19.3.2.2. Multiphase Modeling

Typically, the electrolysis of water produces gaseous hydrogen and oxygen in electrolysis devices, which makes them multiphase fluid systems. The Ansys Fluent mixture multiphase model (see Mixture Model Theory in the *Fluent Theory Guide* (p. 640)) is used in electrolysis and hydrogen pump modeling where the gaseous phase is treated as the primary phase, while liquid water is treated as the secondary phase. The volume fraction of liquid water is obtained from the following equation [583] (p. 1086):

$$\frac{\partial}{\partial t}\left(\alpha_l \varepsilon \rho_l\right)+\nabla\cdot\left(\alpha_l \varepsilon \rho_l \vec{v}_m\right)+\nabla\cdot\left(\vec{i}_m \frac{n_d}{F} M_w\right)=\nabla\cdot\left(-\frac{\rho_l K K_r}{\mu_l}\frac{dp_c}{d\alpha_l}\nabla\alpha_l\right)+S_l+S_{gl} \tag{19.17}$$

where

α_l = volume fraction of liquid water

ε = porosity

ρ_l = liquid water density

$\vec{v}_m$ = velocity of mixture

n_d = osmotic drag coefficient

$\vec{i}_m$ = ionic current density calculated as $\vec{i}_m = -\sigma_m \nabla \phi_m$, where σ_m and ϕ_m are the electrolyte conductivity and electrolyte potential of the membrane, respectively

F = Faraday constant (9.65x10^7 C/kmol)

M_w = molecular weight of liquid water

K = absolute permeability

K_r = relative permeability

μ_l = liquid dynamic viscosity

p_c = capillary pressure

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

S_l = source term for liquid water due to electrochemistry reactions, which depends on electrochemistry reactions and is calculated based on transfer currents

S_{gl} = rate of the mass change between water vapor and liquid water

The third term on the left-hand side of Equation 19.17 (p. 864) represents the effect of electro-osmotic drag, which considers the movement of water through the membrane under the influence of a current carried by ions. Since the flow of membrane zone is not solved, the osmotic drag is treated as a source term in the catalyst layers. In the PEM electrolysis and alkaline electrolysis models, the liquid water is moved due to the movement of ions, while in the H2 pump, the water vapor is moved due to the movement of ions. The osmotic drag coefficient is defined as:

$$n_d=\gamma\frac{2.5}{22}\lambda \tag{19.18}$$

where λ is the water content, and γ is the modified osmotic drag coefficient that is used to generalize the osmotic drag coefficient. In the PEM electrolysis and alkaline electrolysis models, since liquid water is always abundant, the membrane is assumed to be fully hydrated, that is λ=22. In the hydrogen pump, the water content is estimated using an empirical relation as a function of relative humidity r [214] (p. 1065):

$$\lambda=0.043+17.81r-39.85r^2+36.0r^3 \tag{19.19}$$

In electrolysis devices, especially in an H2 pump, sometimes the membrane is not fully hydrated. The membrane hydration level affects electrolyte conductivity. To obtain the membrane hydration level, the transport equation of the water content is solved across the membrane. To account for the effect of the membrane hydration level on the electrolyte conductivity, user-defined functions can be used in Ansys Fluent. To model the water content across the membrane, Ansys Fluent uses the so-called equilibrium approach, which is summarized as follows:

- The water content in the catalyst layers is in equilibrium and is computed from the water activity and water saturation directly based on the empirical formulation [623] (p. 1089):

$$\lambda=\lambda_{eq}=0.3+6a\left(1-\tanh\left(a-0.5\right)\right)+0.69\left(\lambda_{a=1}-3.52\right)a^{0.5}\left(1+\tanh\frac{a-0.89}{0.23}\right)+s\left(\lambda_{s=1}-\lambda_{a=1}\right) \tag{19.20}$$

  where a is the water vapor activity, s is the liquid water saturation, and $\lambda_{s=1}$ and $\lambda_{a=1}$ are the water contents at s=1 and a=1, respectively.

- The water content in the membrane is computed from the diffusion equation:

$$\frac{\partial}{\partial t}\left(\varepsilon M_{w,H_2O}\frac{\rho}{EW}\lambda\right)=\nabla\cdot\left(M_{w,H_2O}D_w\nabla\lambda\right) \tag{19.21}$$

  where ε is the porosity, M_{w,H_2O} is the molecular weight of water, ρ is the water density, EW is the equivalent weight of the membrane, and D_w is the water content diffusivity in the membrane obtained from Equation 21.46 (p. 912).

The transport equations of species are solved for the primary phase. The volumetric source terms (kg/m^3-s) for liquid water, hydrogen, and oxygen due to electrochemistry reactions are calculated

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

at the adjacent cells of the interface. The source terms due to electrochemistry reactions are shown in Table 19.1: Source Terms for Species and Liquid Water due to Electrochemistry Reactions (p. 866).

Table 19.1: Source Terms for Species and Liquid Water due to Electrochemistry Reactions

Component	PEM Electrolysis	Alkaline Electrolysis	Hydrogen Pump
H_2	$S_{H_2}=\frac{M_{w,H_2}}{2F}R_{cat}$		$S_{H_2}=\begin{cases}-\frac{M_{w,H_2}}{2F}R_{an}\\ \frac{M_{w,H_2}}{2F}R_{cat}\end{cases}$
O_2	$S_{O_2}=\frac{M_{w,O_2}}{4F}R_{an}$		N/A
Liquid water	$S_l=-\frac{M_{w,H_2O}}{2F}R_{an}$	$S_l=\begin{cases}\frac{M_{w,H_2O}}{2F}R_{an}\\ -\frac{M_{w,H_2O}}{F}R_{cat}\end{cases}$	

In the above table, M_{w,H_2O}, M_{w,O_2}, and M_{w,H_2} are the molecular mass of water, oxygen, and hydrogen, respectively. R_{an} and R_{cat} are calculated using Equation 21.5 (p. 903) and Equation 21.6 (p. 903).

For alkaline electrolysis and a hydrogen pump, the method to calculate these sources terms is the same, although the formulations are different due to the difference in electrochemical reactions.

The mass transfer rate between water vapor and liquid water is computed based on the diffusion theory:

$$S_{gl}=\begin{cases}\gamma_{cond}\varepsilon(1-s)\frac{(P_v-P_{sat})}{RT} & P_v>P_{sat}\\ \gamma_{evap}\varepsilon s\frac{(P_v-P_{sat})}{RT} & P_v<P_{sat}\end{cases} \tag{19.22}$$

where

ε = porosity

γ_{cond} = condensation rate coefficient

γ_{evap} = evaporation rate coefficient

P_v = water vapor partial pressure

P_{sat} = water saturation pressure

s = liquid water saturation

R = universal gas constant

T = temperature

In an electrolysis device, both the catalyst and the porous layers are often porous media. Therefore, the effects of capillary pressure are also considered in electrolysis simulation. In Ansys Fluent, you can use either the default Leverett function (see Equation 21.35 (p. 909) - Equation 21.37 (p. 909)) or user-defined functions to calculate capillary pressure. In Ansys Fluent, capillary pressure is treated

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

as a diffusion term for the liquid volume fraction as shown the first term on the right-hand side of Equation 19.17 (p. 864) [583] (p. 1086).

For pressurized water electrolysis, typically, the cathode side has higher pressure. As a result, the gas species or liquid water can cross membrane from the higher pressure side to the lower pressure side. To consider such species or liquid water crossover (or permeation), Fick's law is used to estimate the species flux cross the membrane:

$$N = D\frac{\Delta p}{\delta_{mem}} \tag{19.23}$$

where N is the species permeation rate [kg/(m^2 s)], Δp is the pressure difference between the cathode and anode side, δ_{mem} is the thickness of membrane cell zone, and D is the permeation coefficient calculated by:

$$D = \alpha \cdot \exp\left(-\frac{Ea}{RT}\right) \tag{19.24}$$

where α is the permeation rate constant, and Ea is the permeation activation energy. In Ansys Fluent, the volume-averaged pressures on the catalyst layers are used to determine the pressure difference. Since the flow is not solved in the membrane zone, the species permeation is treated as source terms in the first layer of computational cells near the interface between the catalyst layers and membrane.

19.3.2.3. Heat Source

Additional volumetric sources in the thermal energy equation are present because not all electrical work can be converted into chemical energy when producing hydrogen and oxygen. Thermal energy is generated during electrolysis due to electrochemical reactions and irreversibilities of the processes. Volumetric heat source terms in various zones are listed in Table 19.2: Volumetric Heat Source Terms (p. 867).

Table 19.2: Volumetric Heat Source Terms

Zone	Additional Source Term
Current Collector	i_s^2/σ_s
Flow channel	None
Porous Layer	i_s^2/σ_s
Anode Catalyst Layer	$\frac{i_s^2}{\sigma_s}+\frac{i_m^2}{\sigma_m}+R_{an}\left(-\eta_{an}-\frac{T\Delta S_{an}}{nF}\right)$
Cathode Catalyst Layer	$\frac{i_s^2}{\sigma_s}+\frac{i_m^2}{\sigma_m}+R_{cat}\left(-\eta_{cat}-\frac{T\Delta S_{cat}}{nF}\right)$
Membrane	i_m^2/σ_m

In this table:

i_s = solid phase current density

i_m = membrane phase current density

σ_s = conductivity of the solid phase

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

η_{an} = surface overpotential given by Equation 21.11 (p. 904)

η_{cat} = surface overpotential given by Equation 21.12 (p. 904)

T = temperature

n = number of electrons involved in the electrochemistry reactions

ΔS_{an} and ΔS_{cat} = entropy change in Equation 21.13 (p. 904) and Equation 21.14 (p. 904), respectively

19.3.3. Unresolved 0D Modeling Approach

In the unresolved 0D modeling approach, the following transport equation for electric field potential is solved:

$$\nabla \cdot \left(\sigma \nabla \phi \right) = 0 \tag{19.25}$$

The electric field and the electrochemistry interact solely at the electrolyte interface. Ansys Fluent treats the electrolyte interface as an impermeable wall. The potential field must have a "jump" condition applied to the two sides of this wall to account for the effect of the electrochemistry. To closely couple the electrochemical behavior to the potential field calculation, all the electrochemical effects (the voltage jump due to Nernst, the voltage reduction due to activation, and the Ohmic losses due to the resistivity of the electrolyte) must be included in the jump condition. The jump condition is linearized for the voltage reduction due to activation. This interface condition relates the potential on the anode side and the cathode side of the electrolyte and has the following form:

$$\phi_{jump} = U_0 + \eta_{ele} + \left| \eta_{an} \right| + \left| \eta_{cat} \right| \tag{19.26}$$

where η_{an} and η_{cat} are the activation overpotential of the anode and the cathode, respectively, U_0 is the open circuit potential, and η_{ele} is the ohmic overpotential of the electrolyte calculated by:

$$\eta_{ele} = \frac{\delta_{ele}}{\sigma_{ele}} \tag{19.27}$$

where δ_{ele} and σ_{ele} are the thickness and the electrical conductivity of the membrane zone, respectively.

To estimate the activation overpotential of the anode and cathode (η_{an} and η_{cat}), the Butler-Volmer equations are used:

$$j_{an} = \delta_{\text{an}} \xi_{an} j_{an}^{ref} \left(T \right) \left(\frac{\left[C \right]}{\left[C \right]^{ref}} \right)^{\gamma_{an}} \left\{ \exp\left(\frac{\alpha_{an}^{an} F \eta_{an}}{RT} \right) - \exp\left(-\frac{\alpha_{cat}^{an} F \eta_{an}}{RT} \right) \right\} \tag{19.28}$$

$$j_{cat} = \delta_{cat} \xi_{cat} j_{cat}^{ref} \left(T \right) \left(\frac{\left[C \right]}{\left[C \right]^{ref}} \right)^{\gamma_{cat}} \left\{ -\exp\left(\frac{\alpha_{an}^{cat} F \eta_{cat}}{RT} \right) + \exp\left(-\frac{\alpha_{cat}^{cat} F \eta_{cat}}{RT} \right) \right\} \tag{19.29}$$

where

subscripts an and cat denote the anode and cathode zones, respectively

$[\ ]$, $[\ ]_{ref}$ denote the local species concentration and the reference value for the local species concentration, respectively (kmol/m^3)

δ = catalyst layer thickness (m)

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

ξ = specific active surface area (1/m)

$j^{ref}(T)$ = reference exchange current density per active surface area (A/m^2)

γ = concentration dependence

α_{an}^{an} and α_{cat}^{an} = anode and cathode transfer coefficients of the anode electrode, respectively (dimensionless)

α_{an}^{cat} and α_{cat}^{cat} = anode and cathode transfer coefficients of the cathode electrode, respectively (dimensionless)

η = surface overpotential

F = Faraday constant (9.65x10^7 C/kmol)

R = the universal gas constant

T = temperature

After solving the potential equation, the current density generated at the electrolyte interface can be obtained. Based on the current density, the volumetric source and sink terms for each species and energy equations in the adjacent computational cells can be calculated.

The volumetric source terms (kg/m^3-s) for liquid water, hydrogen, and oxygen due to electrochemistry reactions are calculated at the adjacent cells of the interface. The source terms due to electrochemistry reactions are shown in Table 19.3: Source Terms for Species and Liquid Water due to Electrochemistry Reactions in the Unresolved 0D Approach (p. 869):

Table 19.3: Source Terms for Species and Liquid Water due to Electrochemistry Reactions in the Unresolved 0D Approach

Component	PEM Electrolysis	Alkaline Electrolysis	Hydrogen Pump
H_2	$S_{H2}=\frac{M_{w,H2}}{2F}\frac{jA}{V}$		$S_{H2}=\begin{cases}-\frac{M_{w,H2}}{2F}\frac{jA}{V}\\ \frac{M_{w,H2}}{2F}\frac{jA}{V}\end{cases}$
O_2	$S_{O2}=\frac{M_{w,O2}}{4F}\frac{jA}{V}$		N/A
Liquid water	$S_l=-\frac{M_{w,H2O}}{2F}\frac{jA}{V}$	$S_l=\begin{cases}\frac{M_{w,H2O}}{2F}\frac{jA}{V}\\ -\frac{M_{w,H2O}}{F}\frac{jA}{V}\end{cases}$	

In the above table, $M_{w,H2O}$, $M_{w,O2}$, and $M_{w,H2}$ are the molecular mass of water, oxygen, and hydrogen, respectively, j is the current density at the electrolyte interface, A is the area of the electrolyte interface, and V is volume of the adjacent computational cell.

The heat source due to electrochemistry reactions in the adjacent computational cells is calculated as:

$$Q=\begin{cases}\frac{jA}{V}\left(\eta_{an}-\frac{T\Delta S_{an}}{nF}\right)\\ \frac{jA}{V}\left(-\eta_{cat}-\frac{T\Delta S_{cat}}{nF}\right)\end{cases} \tag{19.30}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

where T is temperature, n is the number of electrons involved in the electrochemistry reactions, and ΔS_{an} and ΔS_{cat} are the entropy changes in Equation 21.13 (p. 904) and Equation 21.14 (p. 904), respectively.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Chapter 20: Battery Model

This chapter covers the theory behind the battery models available in Ansys Fluent. For information about how to use battery models, see Modeling Batteries in the Fluent User's Guide.

This chapter describes:

20.1. Battery Solution Methods

20.2. Electro-Chemical Models

20.3. MSMD Solution Method: Coupling Between CFD and Submodels

20.4. Simulating Battery Pack Using the MSMD Solution Method

20.5. Reduced Order Solution Method (ROM)

20.6. External and Internal Electric Short-Circuit Treatment

20.7. Thermal Abuse Model

20.8. Battery Life Models

20.9. Battery Swelling Model

20.1. Battery Solution Methods

In Ansys Fluent, the battery model mainly focuses on conducting thermal analysis of a battery cell or battery pack. During normal operation, a battery produces heat. To compute the heat generation rate, a coupled thermal-electrochemical simulation is used. Depending on how thermal and electro-chemical behaviors of a battery are coupled, FLUENT provides four different solution methods:

- CHT coupling method
- FMU-CHT coupling method
- Circuit Network solution method
- MSMD solution method

In the first two methods, the electro-chemical aspect of a battery system is not resolved in Ansys Fluent. Rather, the heat generation rate is either specified directly or obtained in other software through using the functional mockup unit (FMU).

In the last two methods, the electro-chemical aspect of the battery is solved in Ansys Fluent.

Theoretical information about these methods is presented in the following sections:

20.1.1. CHT Coupling Method

20.1.2. FMU-CHT coupling method

20.1.3. Circuit Network Solution Method

20.1.4. MSMD Solution Method

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

20.1.1. CHT Coupling Method

The Conjugate Heat Transport (CHT) coupling method is the simplest solution method. In this approach, you already have a good knowledge of the battery, and you know the heat generation rate during its operation. In the thermal analysis, the heat source in the energy equation is specified explicitly.

Joule heating in passive conductive zones including battery tabs and busbar connectors can be considered. A single potential equation is solved in the passive zones to obtain the Joule heating. In summary, the following equations are solved in Fluent:

$$\frac{\partial \rho C_p T}{\partial t} - \nabla \cdot \left(k \nabla T \right) = \sigma \left| \nabla \phi \right|^2 + \dot{q}_{bat} + \dot{q}_{abuse} \tag{20.1}$$

$$\nabla \cdot \left(\sigma \nabla \varphi \right) = 0 \tag{20.2}$$

where

σ = electric conductivity of the passive material

$\dot{q}_{bat}$ = battery's heat generation source

The remaining nomenclature is the same as in MSMD Solution Method (p. 873). The first term in the equation is the Joule heating term, which is only added in the passive conductive zones. The battery heat generation term $\dot{q}_{bat}$ is only added in the battery active zones.

20.1.2. FMU-CHT coupling method

Battery cell manufacturers and some original equipment manufacturers (OEMs) may have their own electrochemical models for the electrical performance of their batteries. The FMU-CHT coupling method provides you with the flexibility to use your own electrochemical models, especially your own equivalent circuit-based models (ECMs) by importing it into Ansys Fluent through FMUs.

In this method, Joule heating can be included. The set of equations solved by Ansys Fluent is the same as in the CHT coupling method (see CHT Coupling Method (p. 872)). The only difference is that the heat generation rate is computed from the FMU during a simulation, instead of being specified directly.

20.1.3. Circuit Network Solution Method

In the Circuit Network method, every battery is treated as an electric circuit, and a battery pack is treated as a circuit network. The electric behavior of the battery, including the current and voltage distributions, is computed in a circuit network, and the heat generation rate within each battery can be obtained. Just as in the previous two approaches, the heat generation within a battery is assumed to be uniform. Therefore, the total heat for each battery is distributed uniformly within its active zone. Note that even though the heat generation is assumed to be constant within a battery cell, different battery cells may generate different heat. Moreover, temperature within each battery cell has a non-uniform distribution.

The Joule heating effect within battery passive zones can be optionally included in the simulation. The set of equations solved is the same as in the CHT and FMU-CHT coupling methods. The heat generation is computed from the circuit network calculation.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

20.1.4. MSMD Solution Method

The difficulty with modeling a lithium-ion (Li-ion) battery is due to its multi-domain, multi-physics nature. Vastly different length scales associated with different physics complicates the problem. When performing a thermal analysis, the goal is to determine the temperature distribution at the battery length scale. The physics governing the Li-ion transport occurs in the anode-separator-cathode sandwich layers (the electrode pair length scale). Li-ion transport in an active material occurs at the atomic length scale. The Multi-Scale Multi-Domain (MSMD) approach deals with different physics in different solution domains [[310] (p. 1071)].

Battery thermal and electrical fields are solved in the CFD domain at the battery cell's scale using the following differential equations:

$$\frac{\partial \rho C_p T}{\partial t} - \nabla \cdot \left(k \nabla T \right) = \sigma_+ \left| \nabla \varphi_+ \right|^2 + \sigma_- \left| \nabla \varphi_- \right|^2 + \dot{q}_{ECh} + \dot{q}_{short} + \dot{q}_{abuse} \tag{20.3}$$

$$\nabla \cdot \left(\sigma_+ \nabla \varphi_+ \right) = - \left(j_{ECh} - j_{short} \right)$$
$$\nabla \cdot \left(\sigma_- \nabla \varphi_- \right) = j_{ECh} - j_{short} \tag{20.4}$$

where σ_+ and σ_- are the effective electric conductivities for the positive and negative electrodes, φ_+ and φ_- are phase potentials for the positive and negative electrodes, j_{ECh} and $\dot{q}_{ECh}$ are the volumetric current transfer rate and the electrochemical reaction heat due to electrochemical reactions, respectively, j_{short} and $\dot{q}_{short}$ are the current transfer rate and heat generation rate due to battery internal short-circuit, respectively, and $\dot{q}_{abuse}$ is the heat generation due to the thermal runaway reactions under the thermal abuse condition.

For normal operation, $\dot{q}_{abuse}$ is set to zero. For more information, refer to Thermal Abuse Model (p. 885). The source terms j_{ECh} and $\dot{q}_{ECh}$ are computed using an electrochemical submodel. If there is no internal short-circuit, j_{short} and $\dot{q}_{short}$ are equal to zero. For more information, refer to External and Internal Electric Short-Circuit Treatment (p. 884).

A wide range of electrochemical models, from simple empirically-based to fundamental physics-based, is available in the open literature. In Ansys Fluent, the following electrochemical submodels are implemented:

- Newman, Tiedemann, Gu and Kim (NTGK) or direct current internal resistance (DCIR) model
- Equivalent Circuit Model (ECM) model
- Newman Pseudo-2D (P2D) model

In addition, you can define your own electrochemical model via user-defined functions and hook it to the Fluent MSMD battery module.

To use the MSMD approach in an arbitrary finite volume of the cell composite, the following conditions must be met:

- Battery micro layers must have the same orientation
- Two potential fields, φ_+ and φ_-, must be uniquely determined

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

These conditions are usually met for aligned stack cells or wound cells with extended foil-type continuous current tabs.

The MSMD solution method can resolve the potential distribution within a battery and, therefore, the detailed heat generation distribution within the battery. Since two extra transport equations are solved, the computational cost is higher compared to the previous methods.

20.2. Electro-Chemical Models

In the Circuit Network and the MSMD solution methods, the electric aspect of a battery is solved in Ansys Fluent. The following electro-chemical submodels are available in Ansys Fluent:

- NTGK/DCIR Model
- ECM Model
- Newman's P2D Model

The NTGK/DCIR and ECM models are semi-empirical, while the Newman's P2D model is physics-based.

Details about these models are presented in the following sections:

20.2.1. NTGK/DCIR Model

20.2.2. ECM Model

20.2.3. Newman's P2D Model

20.2.1. NTGK/DCIR Model

The Newman, Tiedemann, Gu, and Kim (NTGK) and direct current internal resistance (DCIR) models are two simple semi-empirical electrochemical models that belong to the same group. The NTGK model was proposed by Kwon [[328] (p. 1072)] and has been used by others [[312] (p. 1071), [311] (p. 1071)]. In the model formulation, the volumetric current transfer rate in Equation 20.4 (p. 873) is related to the potential field by the following algebraic equation:

$$j_{ECh} = \frac{Q_{nominal}}{Q_{ref} Vol} Y\left[U - V\right] \tag{20.5}$$

where Vol denotes the active zone's volume of a single battery; V is the battery cell voltage, which is either obtained directly from the circuit network solution method or calculated as φ_+-φ_- from the MSMD solution method; $Q_{nominal}$ is the battery total electric capacity in Ampere hours; and Q_{ref} is the capacity of the battery that is used in experiments to obtain the model parameters Y and U. Y and U are functions of the battery depth of discharge (DoD):

$$DoD = \frac{Vol}{3600 Q_{nominal}} \int_0^t j \, dt \tag{20.6}$$

For a given battery, the voltage-current response curve can be obtained through experimentation. Then Y and U in Equation 20.5 (p. 874) can be fitted, in theory. There are two approaches to specifying Y and U functions:

- Y and U can be fitted from testing data as functions of DOD and temperature explicitly before simulations. The relationship between Y/U and DOD/temperature can be provided:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

– through function forms as proposed in [328] (p. 1072)

$$U=\left(\sum_{n=0}^{5}a_n\left(DoD\right)^n\right)-C_2\left(T-T_{ref}\right) \tag{20.7}$$

$$Y=\left(\sum_{n=0}^{5}b_n\left(DoD\right)^n\right)\exp\left[-C_1\left(\frac{1}{T}-\frac{1}{T_{ref}}\right)\right]$$

where C_1 and C_2 are the battery-specific NTGK model constants.

– by a two-dimensional lookup table

- Y and U are not determined in advance. Instead, the relationship of current and voltage from raw test data is stored and Y and U are computed on-the-fly during simulation.

Model parameters are battery specific. You can use the Y and U parameters from a tested battery with capacity Q_{ref} for a battery with different capacity $Q_{nominal}$ as long as the battery's material is the same. If the model parameters are from the battery that is analogous to the one you are simulating, then capacities $Q_{nominal}$ and Q_{ref} will be the same.

In the formulation of the NTGK model, $1/Y$ can be regarded as battery internal resistance. In applications where internal resistance is known, it can be defined directly in Ansys Fluent. See Inputs for the NTGK/DCIR Model in the *Fluent User's Guide* for more information.

The electrochemical reaction heat $\dot{q}_{ECh}$ in Equation 20.3 (p. 873)) is calculated as

$$\dot{q}_{ECh}=j_{ECh}\left[U-V-T\frac{dU}{dT}\right] \tag{20.8}$$

where the first term is heat due to overpotential and the second term is heat due to entropic heating.

For conditions where the electric load does not change much, the NTGK/DCIR model can be used for simulating the battery charging / discharging cycle. However, should the sudden surges in the electric load occur, such as in a real driving cycle, the NTGK/DCIR method will not account for the inertial changes. For simulations of such abrupt fluctuations in the electric load, you can use the Ansys Fluent Equivalent Circuit Model (ECM) model. See ECM Model (p. 875) for details about the ECM model.

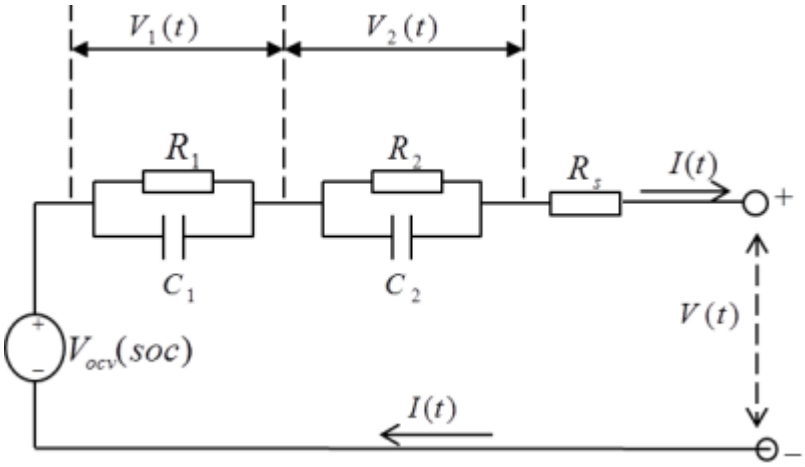
For more information about setting up the NTGK/DCIR model, see Inputs for the NTGK/DCIR Model.

20.2.2. ECM Model

In the Equivalent Circuit Model (ECM), battery electric behavior is mimicked by an electrical circuit. Ansys Fluent has adopted the six parameter ECM model following the work of Chen [[105] (p. 1059)]. In this model, the circuit consists of three resistors and two capacitors (Figure 20.1: Electric Circuits Used in the ECM Model (p. 876)).

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Figure 20.1: Electric Circuits Used in the ECM Model



The voltage-current relation can be obtained by solving the electric circuit equations:

$$V = V_{OCV}(soc) + V_1 + V_2 - R_s(soc)I(t) \tag{20.9}$$

$$\frac{dV_1}{dt} = -\frac{1}{R_1(soc)C_1(soc)}V_1 - \frac{1}{C_1(soc)}I(t)$$

$$\frac{dV_2}{dt} = -\frac{1}{R_2(soc)C_2(soc)}V_2 - \frac{1}{C_2(soc)}I(t)$$

$$\frac{d(soc)}{dt} = I(t)\Big/3600Q_{ref}$$

Here, V is the battery cell voltage that can be either obtained directly from the circuit solution in the Circuit Network solution method or calculated as φ_+-φ_- from the MSMD solution method. For a given battery, the open circuit voltage, resistors' resistances, and capacitors' capacitances are functions of the battery state of charge (SOC) and temperature. These functions could be expressed in three different ways in Ansys Fluent:

- The fifth order Polynomial form:

$$R_s = a_0 + a_1(soc) + a_2(soc)^2 + a_3(soc)^3 + a_4(soc)^4 + a_5(soc)^5 \tag{20.10}$$

$$R_1 = b_0 + b_1(soc) + b_2(soc)^2 + b_3(soc)^3 + b_4(soc)^4 + b_5(soc)^5$$

$$C_1 = c_0 + c_1(soc) + c_2(soc)^2 + c_3(soc)^3 + c_4(soc)^4 + c_5(soc)^5$$

$$R_2 = d_0 + d_1(soc) + d_2(soc)^2 + d_3(soc)^3 + d_4(soc)^4 + d_5(soc)^5$$

$$C_2 = e_0 + e_1(soc) + e_2(soc)^2 + e_3(soc)^3 + e_4(soc)^4 + e_5(soc)^5$$

$$V_{ocv} = f_0 + f_1(soc) + f_2(soc)^2 + f_3(soc)^3 + f_4(soc)^4 + f_5(soc)^5$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

- The function form proposed by Chen [[105] (p. 1059)]:

$$\begin{aligned} R_s &= a_0 + a_1 \exp\left[-a_2(soc)\right] \\ R_1 &= b_0 + b_1 \exp\left[-b_2(soc)\right] \\ C_1 &= c_0 + c_1 \exp\left[-c_2(soc)\right] \\ R_2 &= d_0 + d_1 \exp\left[-d_2(soc)\right] \\ C_2 &= e_0 + e_1 \exp\left[-e_2(soc)\right] \\ V_{ocv} &= f_0 + f_1(soc) + f_2(soc)^2 + f_3(soc)^3 + f_4 \exp\left[-f_5(soc)\right] \end{aligned} \tag{20.11}$$

- 2D table

  This method allows to account for the temperature effect, which cannot be considered in the above two methods. It uses a two dimensional structured table to define each parameter as a function of SOC and temperature.

The source terms for Equation 20.3 (p. 873) and Equation 20.4 (p. 873) are computed as:

$$j_{ECh} = I \frac{Q_{Nominal}}{Q_{ref} Vol} \tag{20.12}$$

$$\dot{q}_{ECh} = j_{ECh}\left[V_{ocv} - V - T\frac{dU}{dT}\right] \tag{20.13}$$

where I is the current, and V_{ocv} is the open circuit voltage.

Similar to the NTGK/DCIR model, you can use model parameters from a tested battery with capacity Q_{ref} for a battery with different capacity $Q_{nominal}$ as long as the battery's material is the same. If the model parameters are from the battery that is analogous to the one you are simulating, then capacities $Q_{nominal}$ and Q_{ref} will be the same.

Optionally, you can account for the battery property dependencies on other factors through user-defined functions. See Battery Model `DEFINE` Macros in the *Fluent Customization Manual* for details.

The battery parameters and their dependencies on SoC and temperature can be either obtained from a battery OEM vendor or evaluated from a physical prototype by using the parameter estimation tool provided by Ansys Fluent. The tool is available via the text user interface (TUI) (see Battery Model Text Commands in the *Fluent Text Command List*). The parameter estimation tool can be used to accept the recorded inputs from the more computationally expensive Newman's P2D model.

The ECM model allows you to simulate battery discharging and recharging processes. The ECM model is economical, versatile, and suitable for simulating both a single battery cell and a battery pack. It can be used for various types of solid batteries, not only for Li-ion types.

For more information about setting up the ECM Model, see Inputs for the Equivalent Circuit Model.

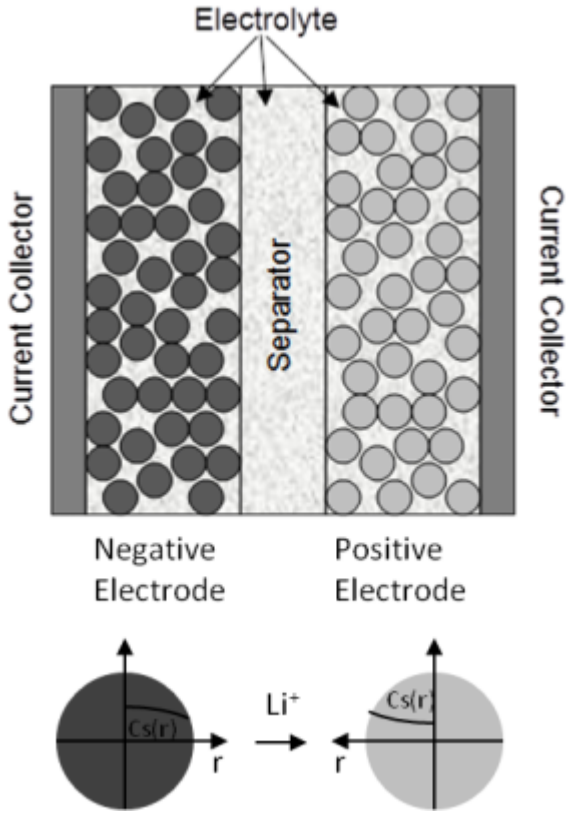
20.2.3. Newman's P2D Model

Newman's group developed a physics based model using porous electrode and concentrated solution theories [[148] (p. 1061)]. The model can accurately capture Li-ion migration in the battery. It has been widely used in the literature [[609] (p. 1088), [90] (p. 1058)].

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Figure 20.2: Electrode and Particle Domains in the Newman's Model (p. 878) schematically shows the electrode plate pair in the Li-ion battery. The composite electrode consists of active materials and electrolyte solution. The electrolyte phase is continuous across the negative electrode, separator and positive electrode while solid phase only exists in the negative and positive electrode. The solid active material is modeled as a matrix of mono-sized spherical particles.

Figure 20.2: Electrode and Particle Domains in the Newman's Model



During the discharge process, Li diffuses to the surface of negative electrode particles and undergoes an electrochemical reaction. This reaction releases an electron and transfers Li to the electrolyte phase. The Li-ions diffuse and conduct through the electrolyte solution from the negative electrode to the positive electrode, where a similar reaction transfers Li to the positive solid phase. Li is stored inside the positive electrode particles until the cell is later recharged.

The Li-ion transport phenomena in the porous electrode and electrolyte can be described by the charge and mass conservation laws. Charge conservation governs phase potentials, φ_+ and φ_-, while mass conservation governs the phase concentrations, c_e and c_s, where the subscripts e and s are used to denote the electrolyte and solid (electrode) phases, respectively.

The conservation equations relevant to electrolyte and solid phases are described as:

- Lithium conservation in the solid phase:

$$\frac{\partial c_s}{\partial t} = \frac{D_s}{r^2}\frac{\partial}{\partial r}\left(r^2\frac{\partial c_s}{\partial r}\right) \tag{20.14}$$

- Lithium conservation in the electrolyte phase:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$\frac{\partial(\varepsilon_e c_e)}{\partial t} = \frac{\partial}{\partial x}\left(D_e^{eff}\frac{\partial c_e}{\partial x}\right) + \frac{a_s(1-t_+^0)}{F}\left(j^{int} + \sum j^{side}\right) \tag{20.15}$$

- Charge conservation in solid phase:

$$\frac{\partial}{\partial x}\left(\sigma^{eff}\frac{\partial \varphi_s}{\partial x}\right) - a_s\left(j^{int} + \sum j^{side}\right) = 0 \tag{20.16}$$

- Charge conservation in the electrolyte phase:

$$\frac{\partial}{\partial x}\left(\kappa^{eff}\frac{\partial \varphi_e}{\partial x}\right) + \frac{\partial}{\partial x}\left(\kappa_d^{eff}\frac{\partial \ln c_e}{\partial x}\right) + a_s\left(j^{int} + \sum j^{side}\right) = 0 \tag{20.17}$$

The Lithium conservation diffusion equation must be solved at every discretized spatial location in the electrode zone. The Lithium conservation equation is solved in the r-dimension of the spherical particles—the pseudo second dimension. That is why the model is often referred to as the Newman's P2D model in the literature.

The Butler-Volmer equation is used to couple the charge and species governing equations by describing the current flux at the active material pore-wall surface due to lithium intercalation reaction j^{int} as a function of overpotential, η:

$$j^{int} = i_0\left\{\exp\left(\frac{\alpha_a F}{RT}\eta\right) - \exp\left(-\frac{\alpha_c F}{RT}\eta\right)\right\} \tag{20.18}$$

where η is defined by

$$\eta = \varphi_s - \varphi_e - U - R_{film}\left(j^{int} + \sum j^{side}\right) \tag{20.19}$$

and i_0 is the exchange current density defined by

$$i_0 = F k_m (c_e)^{\alpha_a}\left(C_{s,max} - c_{s,e}\right)^{\alpha_a}\left(c_{s,e}\right)^{\alpha_c} \tag{20.20}$$

The last term on the right-hand side of Equation 20.19 (p. 879) is the potential drop due to the electric resistance of the electrode particle surface film.

Effective properties are used in the above equations in which the electrolyte ionic diffusivity, conductivity, diffusional conductivity, and solid phase electric conductivity are defined as:

$$D_e^{eff} = D_e \varepsilon_e^{\beta} \tag{20.21}$$

$$\kappa^{eff} = \kappa \varepsilon_e^{\beta} \tag{20.22}$$

$$\kappa_D^{eff} = \frac{2RT\kappa^{eff}}{F}\left(t_+^0 - 1\right)\left(1 + \frac{d\ln f_{\pm}}{d\ln c_e}\right) \tag{20.23}$$

$$\sigma^{eff} = \sigma \varepsilon_s^{\beta} \tag{20.24}$$

$$a_s = 3\varepsilon_s / r_s \tag{20.25}$$

$$D_s = D_{s,ref} exp\left[-E_d / R\left(1/T - 1/T_{ref}\right)\right] \tag{20.26}$$

$$k_m = k_{m,ref} exp\left[-E_r / R\left(1/T - 1/T_{ref}\right)\right] \tag{20.27}$$

Note that the sum of the volume fractions of electrolyte and active and filler materials must sum to unity:

$$\varepsilon_e + \varepsilon_f + \varepsilon_s = 1 \tag{20.28}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

The above equations use the following notations:

σ^{eff} is the effective electric conductivity

j^{int} is the active material pore-wall current flux due to lithium intercalation reaction

j^{side} is the active material pore-wall current flux due to side reactions

κ is the electrolyte ionic conductivity

κ_D^{eff} is the electrolyte diffusional conductivity

D_s is the diffusion coefficient of Li in solid

D_e is the diffusion coefficient of Li+ in the electrolyte phase

t_+^0 is the transference number of lithium ion

U is the open circuit potential of an electrode reaction, which is usually a function of state of lithiation, $c_s/c_{s,\max}$

$c_{s,\max}$ is the maximum concentration of lithium in solid phase

$c_{s,e}$ is the concentration of lithium at the surface of solid particles

α_a is the charge transfer coefficient at anode

α_c is the charge transfer coefficient at cathode

a_s is the solid/electrolyte interfacial area per unit volume

F is the Faraday constant

R is the universal gas constant

R_{film} is the electric resistance of the electrode particle surface film

$f_{\pm}$ is the electrolyte activity coefficient

β is the Bruggeman porosity exponent

ε_e is the volume fraction of the electrolyte phase in electrode

ε_s is the volume fraction of active material in electrode

ε_f is the volume fraction of the filler material in electrode

$D_{s,ref}$ is the reference solid diffusion coefficient

$k_{m,ref}$ is the reference reaction rate constant

E_d is the activation energy that controls the temperature sensitivity of D_s

E_r is the activation energy that controls the temperature sensitivity of k_m

T_{ref} is the reference temperature, 298 K

The source terms for the potential equations (Equation 20.4 (p. 873)) and the energy equation (Equation 20.3 (p. 873)) is computed as:

$$j_{ECh} = -\frac{Q_{nominal}}{Q_{ref} Vol} i_P \tag{20.29}$$

$$\dot{q}_{ECh} = \frac{i_P V + \int_{l_p+l_s+l_n}^{0} j^{Li}\left(T_{ref}\frac{\partial U}{\partial T} - U_{Ref}\right)dx}{l_p+l_s+l_n} \tag{20.30}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

where l_p, l_n, and l_s are the thicknesses of positive electrode, negative electrode, and separator, respectively, and i_P is the transverse current density

$$i_P = \int_0^{l_p} j^{Li} dx \tag{20.31}$$

In practice, the Newman's P2D model is solved on the electrode scale. Q_{ref} is usually based on the unit area of electrode. Equation 20.29 (p. 880) is used to connect the current transfer rate at the battery cell scale to the electrode scale.

For more information about setting up the Newman's P2D Model, see Inputs for the Newman's P2D Model.

20.3. MSMD Solution Method: Coupling Between CFD and Submodels

In theory, an electrochemical submodel could be constructed and solved for every CFD cell at every flow iteration (for steady simulations) or timestep (for transient simulations). For the ECM model, this results in solving a set of four algebraic differential equations. For the Newman model, this can involve solving a system of hundreds of algebraic differential equations, depending upon the number of discretization points used in the electrode and particle domain. This adds to a tremendous computational cost of the simulation. In practice, the submodel, especially the Newman's P2D model, is typically constructed for a group of cells, called a cluster.

The idea of cell clustering comes from the fact that the current transfer rate, j, is more or less uniform, although current itself is not. As a result, instead of solving the submodel for every computational cell, we can group CFD cells into clusters and solve the submodel only once for each cluster.

Ansys Fluent provides a tool for grouping the CFD cells into clusters. For a user-specified number of clusters N_x, N_y, and N_z in the X, Y, and Z directions respectively, the solver will divide the battery electrochemical domain into $N_x N_y N_z$ clusters and then obtain the solution for each cluster using the cluster-average values.

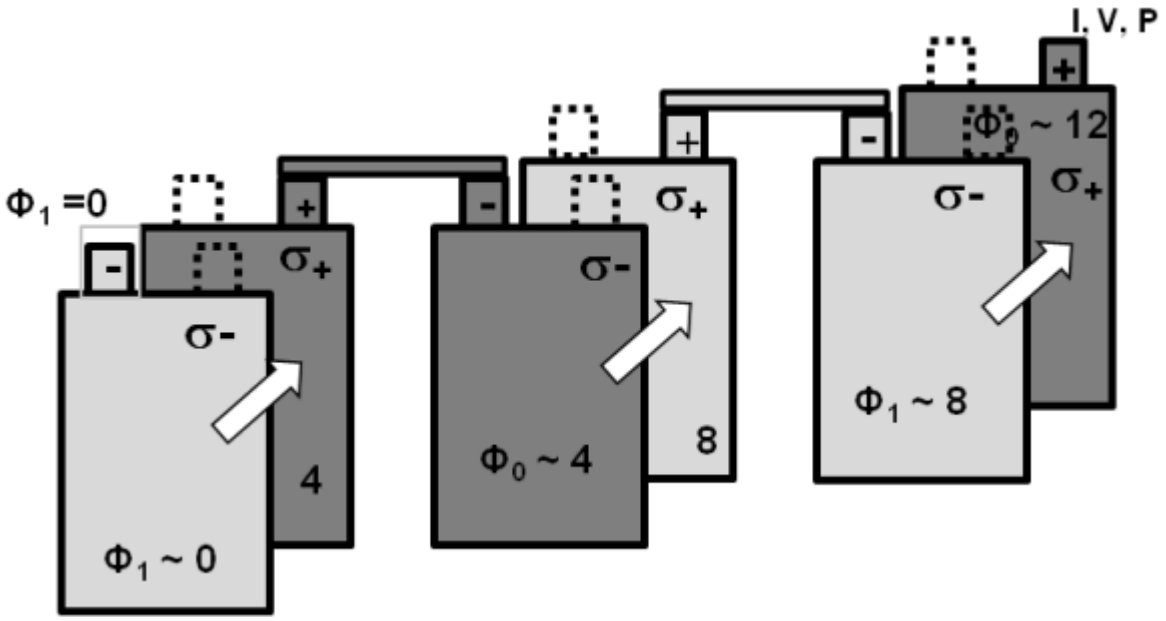
20.4. Simulating Battery Pack Using the MSMD Solution Method

Many applications utilize battery systems in which multiple batteries are connected in series or in parallel. In Ansys Fluent, the MSMD approach has been extended to simulate battery systems. When simulating a battery pack, Ansys Fluent uses the continuous zone method in which two potential equations are solved over all the conductive zones continuously. In a single battery case, two potential equations can be interpreted as φ_+ and φ_-, where φ_+ is the potential field of the positive electrode and φ_- is the potential of the negative electrode. In a multiple cell system, the situation is more complicated as explained below.

Figure 20.3: Solution Domain for Two Potential Equations in a Battery Pack System (p. 882) schematically shows an example in which three battery cells are connected in series.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Figure 20.3: Solution Domain for Two Potential Equations in a Battery Pack System



The positive tab of one battery is connected to the next battery's negative tab. As a result, the φ_+ of the first battery is about at the same level as the φ_- of the second battery. In other words, φ_+ of the first battery and φ_- of the second battery should be solved in the same domain. We still can use only two potential fields to capture the electric field, however, we can no longer interpret them as the potential fields of the positive and negative electrodes.

In Figure 20.3: Solution Domain for Two Potential Equations in a Battery Pack System (p. 882), different shades of gray are used for cell zones to demonstrate the different equations being solved (for the convenience of bookkeeping, we introduce the "level" of a zone to mark which equation needs to be solved for a given zone):

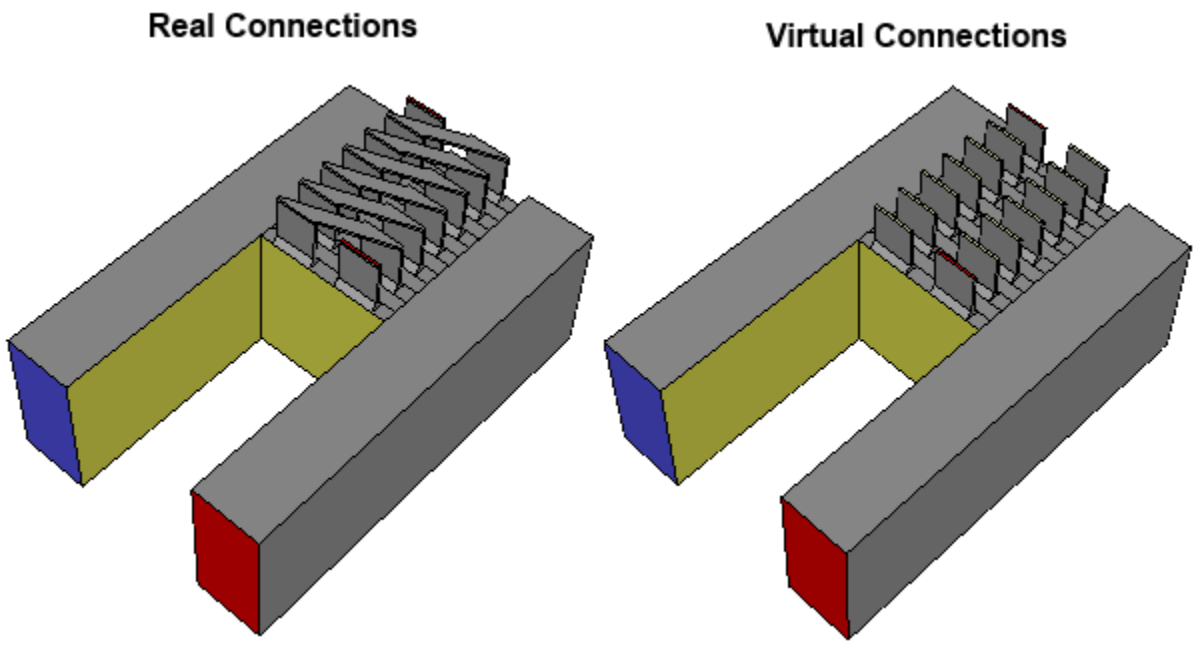
- An active zone (where electro-chemical reactions occur) belongs to two "levels". Both φ_1 and φ_2 are solved: φ_1 is solved for the zones at the odd level and φ_2 is solved in the zones at the even level.
- A passive zone (tab or busbar) belongs to only one level, where only one potential equation is solved.

For example, for N batteries connected in series, there will be N+1 levels. Except for the first and the last levels, each level is isolated in the sense that there is no flux across its boundary. Two levels are coupled only through the equations source terms. Two potential equations are solved continuously across all the conductive zones. In a passive zone, however, only one potential equation is solved, and the second equation is excluded by setting the electrical conductivity to zero.

Battery Connections

Batteries in a pack can be connected through either "real connections" or "virtual connections".

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.



Real connections are busbars that are physically resolved and meshed in the model.

Virtual connections are busbars or battery tab volumes that are not explicitly resolved in the model. Instead, the connection information is provided in a battery connection definition file as described in Specifying Electric Contacts. The solver reads this information and sets the electric boundary conditions for each individual battery through the virtual connections.

Using virtual connections saves the time and effort of meshing busbar and tab volumes, which are usually very thin and difficult to mesh. The disadvantage is that electric resistance and Joule heating are not considered in these unresolved volumes.

You can select the method that best suits your needs in your simulation. Refer to Specifying Electric Contacts for the details on using the battery virtual connections in a battery pack.

20.5. Reduced Order Solution Method (ROM)

The MSMD method needs to solve two potential equations (Equation 20.4 (p. 873)). Although the equations do not have transient terms, they are solved repeatedly during each time step in a transient fashion due to the time-varying electric load condition and the change of state of a battery charge.

However, under the following two conditions, the solution of these two potential equations can be easily found without solving them repeatedly:

- Electric conductivity does not depend on temperature
- Transfer current density is uniform over a battery's active zone

A careful examination of Equation 20.4 (p. 873) and their boundary conditions reveals that the two equations become homogeneous linear partial differential equations under those two conditions. Instead of solving them at each time step, a scaling procedure can be used, so that the two potential fields under any electric load condition can be obtained from a reference potential field. The reference potential field can be pre-calculated at a reference load condition. By using this method, the need to solve the

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

two potential equations at each CFD time step is eliminated entirely. The cost of a general electrochemical-thermal coupled simulation is reduced to that of a pure thermal simulation. This can save a significant amount of computational time in a battery simulation.

20.6. External and Internal Electric Short-Circuit Treatment

The Ansys Fluent battery model can handle both external and internal short-circuits.

External Short-Circuit

In the case of an external short-circuit, battery's positive and negative tabs are connected directly with no electric load. In Ansys Fluent, an electric load boundary condition for the external resistance enables you to directly specify the total external electric resistance, R_{load}. The battery model solver will guarantee that the ratio of the battery's tab voltage, V_{tab}, to the tab current, I_{tab}, equals the specified resistance value:

$$R_{load} = V_{tab} / I_{tab}$$

Internal Short-Circuit

Under normal battery operation, the battery's positive and negative electrodes are divided by a separator, usually a thin polymer material that prevents electrons from traveling directly from the negative electrode to the positive electrode. In the case of an internal short-circuit, which could be a result of a nail penetration or a crash accident, the separator is ruptured in a localized area. Besides providing the normal tab current, the battery produces a secondary electric current from electrochemical reactions that is wasted through the shorted area.

The short-circuit intensity can be described by a fundamental variable-volumetric contact resistance, which could be computed as r_c^0 / a, where r_c^0 is the contact resistance of the electrode sheet in the unit of $\Omega \cdot m^2$, and a is the specific area of the electrode sheet in the battery volume in the unit of 1/m. In the MSMD solution method, the sandwich-style electrode layers are not resolved; instead, they are treated as a homogeneous medium. So, the combined variable—volumetric contact resistance r_c in the unit of $\Omega \cdot m^3$—is used in the calculation. The transfer short current density, j_{short} in Equation 20.4 (p. 873), and extra heat generated due to the short are computed as:

$$\begin{aligned} j_{short} &= j'_{short} = (\phi_+ - \phi_-) / r_c \\ \dot{q}_{short} &= (\phi_+ - \phi_-)^2 / r_c \end{aligned} \tag{20.32}$$

The above formulation is general enough to cover non-short cases where r_c tends to infinity. Contact resistance could be a function of time and location, therefore allowing the dynamic evolution of the changes in the size of the shorted region and the intensity of the short-circuit current.

For information on how to use and set up this feature, see Specifying External and Internal Short-Circuit Resistances.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

20.7. Thermal Abuse Model

Safety has become an important issue in battery design. Under abuse conditions, thermal runaway may occur in a battery. Increased temperature could trigger thermal runaway reactions. Excessive of heat released as a result of such conditions could damage a battery cell, or even cause fire or explosion. Thermal runaway reactions are very complicated and material-specific. For modeling thermal runaway reactions and simulating a battery thermal behavior under thermal abuses, Ansys Fluent offers two semi-empirical models:

- One-Equation Model
- Four-Equation Model

For details on how to set up and use these model, see Specifying Advanced Options.

One-Equation Model

In the one-equation model, thermal runaway reactions are lumped together as one reaction that can be described by the following kinetics equation [403] (p. 1076):

$$\frac{d\alpha}{dt} = A \cdot \exp\left(-E/RT\right)\alpha^{m}\left(1-\alpha\right)^{n} \tag{20.33}$$

where:

α = degree of conversion

A = pre-exponential factor of the reaction (s^{-1})

E = activation energy of the reaction (J/mol)

R = universal gas constant

T = temperature

m and n = reaction order parameters

Equation 20.33 (p. 885) is solved to keep track of the progress of the thermal runaway reaction. α changes from 0 to 1, with 0 denoting "no reaction", and 1 denoting "completion of reaction".

With α computed from Equation 20.33 (p. 885), the heat generation rate due to thermal runaway can be calculated as:

$$\dot{q}_{abuse} = HW \cdot \left|\frac{d\alpha}{dt}\right| \tag{20.34}$$

where HW is the specific heat release (J/m^3).

Four-Equation Model

In the four-equation model, thermal runaway reactions are put into the following four categories [309] (p. 1071):

- solid electrolyte interface (SEI) decomposition reactions
- negative electrode-electrolyte reactions
- positive electrode-electrolyte reactions

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

- electrolyte decomposition reactions

To keep track of the progress of each type of reactions, a lumped Arrhenius kinetics rate is used:

$$\frac{dc_{sei}}{dt} = -A_{sei}\exp\left[-\frac{E_{sei}}{RT}\right]c_{sei}^{m_{sei}} \tag{20.35}$$

$$\frac{dc_{ne}}{dt} = -A_{ne}exp\left(-\frac{t_{sei}}{t_{sei,ref}}\right)\exp\left[-\frac{E_{ne}}{RT}\right]c_{ne}^{m_{ne}} \tag{20.36}$$

$$\frac{d\alpha}{dt} = A_{pe}\exp\left[-\frac{E_{pe}}{RT}\right]\alpha^{m_{pe,1}}\left(1-\alpha\right)^{m_{pe,2}} \tag{20.37}$$

$$\frac{dc_e}{dt} = -A_e\exp\left[-\frac{E_e}{RT}\right]c_e^{m_e} \tag{20.38}$$

where:

A, E, and various m are reaction kinetics parameters, representing pre-exponential factor, activation energy, and reaction order, respectively

subscripts sei, ne, pe, and e denote parameters associated with SEI decomposition reaction, negative electrode-electrolyte reaction, positive electrode-electrolyte reaction, and electrolyte decomposition reaction, respectively

c_{sei}, c_{ne}, α, and c_e are dimensionless variables that can be interpreted as the fraction of the remaining reactant associated to each type of reactions in the medium as reactions proceed

t_{sei} is the dimensionless measure of a SEI layer thickness

$t_{sei,ref}$ is the reference SEI layer thickness

T is the temperature

R is the universal gas constant

The values of c_{sei}, c_{ne}, and c_e change from 1 (representing the state when nothing has reacted yet) to 0 (denoting the state with reactants being completely consumed). α, on the other hand, varies from 0 to 1, as in the One-Equation model.

t_{sei} can be computed as:

$$t_{sei} = t_{sei,0} + \left(c_{neg,0} - c_{neg}\right)$$

where $t_{sei,0}$ and $c_{neg,0}$ are two initial values. Note that the SEI layer is growing as the negative electrode-electrolyte reaction progresses.

The total heat generation due to the thermal runaway reactions (in units of W/m^3) can be computed as:

$$\dot{q}_{abuse} = H_{sei}W_{sei}\left|\frac{dc_{sei}}{dt}\right| + H_{ne}W_{ne}\left|\frac{dc_{ne}}{dt}\right| + H_{pe}W_{pe}\left|\frac{d\alpha}{dt}\right| + H_eW_e\left|\frac{dc_e}{dt}\right| \tag{20.39}$$

where W (kg/m^3) is density of reactants in the medium, and H is the heat of reaction. All heats of reaction are in units of J/kg.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

If the internal short reaction is considered, the heat due to internal short, $H_{ec}\left|\frac{dSoc}{dt}\right|$ where H_{ec} is the heat of reaction [J/m^3], is added to the thermal abuse heat.

Abuse reaction due to the internal short circuit

In both the one-equation and four-equation models, another abuse reaction due to the internal short circuit can be optionally added [120] (p. 1059). In this model, the internal short circuit reaction is governed by the following kinetics rate equation:

$$\frac{dSoc}{dt} = -ISC_{cond} \cdot Soc \cdot A_{ec} \cdot exp\left(-E_{ec}/RT\right) \tag{20.40}$$

where Soc is the progress variable for the internal short reaction, A_{ec} is a pre-exponential factor, and E_{ec} is activation energy of the reaction. The subscript ec refers to the short-circuit electrochemical reaction. The factor ISC_{cond} is used to make sure that the reaction is activated only after the user-specified trigger temperature $T_{trigger}$ is reached:

$$ISC_{cond} = \begin{cases} 0 & \text{for } T < T_{trigger} \\ 1 & \text{for } T \geq T_{trigger} \end{cases} \tag{20.41}$$

20.8. Battery Life Models

The battery capacity changes over time and over repeated discharging and charging cycles. Capacity fading could be caused by many different mechanisms such as overcharging, growth of solid-electrolyte interphase (SEI) due to undesirable side reactions, mechanical cracks, and so on. Ansys Fluent uses both empirical and physics-based models.

For the emperical models, you can consider the battery capacity fade over both long and short time periods. To estimate long-term battery aging, the emperical battery life model is used. The capacity fade effect is treated as a pre-processing procedure in CFD, in which the usable capacity is updated before a CFD simulation. To estimate short-term battery aging, the battery capacity fade model is used. Here, the capacity fade can be specified on-the-fly as a function of temperature and C-rate.

The physics-based model incorporates the fundamental aging mechanisms into the Newman's P2D model. The model allows you to predict the battery aging process during a large number of electric load cycles in the standalone mode. You can further consider the aging effects in CFD simulations as a preprocessing procedure by plugging in the standalone simulation results as steady state profiles.

Details about these models are presented in the following sections:

20.8.1. Empirical Battery Life Model

20.8.2. Battery Capacity Fade Model

20.8.3. Physics-Based Battery Life Model

20.8.1. Empirical Battery Life Model

Battery capacity loss occurs if the battery is in active use when it undergoes repeated discharging and charging cycles. Such loss is called cycle life loss. However, even if the battery is not in use, its capacity may fade due to a long-time storage. Such loss is called calendar life loss.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

In Ansys Fluent, an Arrhenius type equation is used to compute calendar life loss, and a two-dimensional lookup is used for the cycle life loss:

$$Calendar\ life\ loss = A \cdot exp\left[-E\left(1/T - 1/T_o\right)\right] t^{\alpha} \tag{20.42}$$

$$Cycle\ life\ loss = f\left(T, N_{cycle}\right) \tag{20.43}$$

$$CCF = 1 - \left(Calendar\ life\ loss + Cycle\ life\ loss\right) \tag{20.44}$$

where

A = pre-exponential factor

E = activation energy

T = battery average operation temperature

T_o = reference temperature under which the Arrhenius parameter are obtained

t = time

α = time exponent factor

CCF = capacity correction factor

Once CCF is obtained, the battery usable capacity used in CFD can be calculated:

$$Q_{usable} = Q_{nominal} CCF \tag{20.45}$$

20.8.2. Battery Capacity Fade Model

During a CFD simulation, the capacity fade can also be considered for either the NTGK/DCIR or ECM battery model. Ansys Fluent uses an empirical model, in which you can specify different capacity fade factors under different operating conditions (temperature and C-rate). The specified capacity factors are then used in the calculation of the battery state of charge as follows:

- In the NTGK/DCIR model, Equation 20.6 (p. 874) is modified to the following form to consider the capacity fade effect:

$$DoD = \frac{Vol}{3600 Q_{nominal} CCF} \int_0^t j dt \tag{20.46}$$

- In the ECM model, the third equation in Equation 20.9 (p. 876) is replaced by

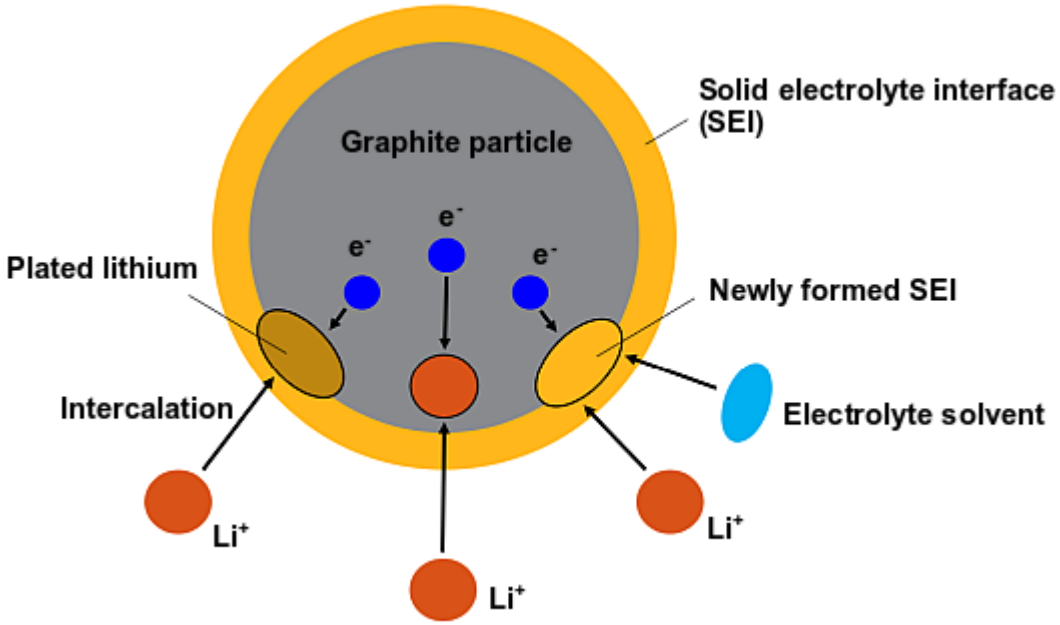
$$\frac{d\left(soc\right)}{dt} = \frac{I\left(t\right)}{\left(3600 Q_{ref} CCF\right)} \tag{20.47}$$

20.8.3. Physics-Based Battery Life Model

In Ansys Fluent, the physics-based battery life model considers the aging mechanisms of the growth of the solid electrolyte interphase (SEI) and the decomposition of metallic lithium (Li plating) [720] (p. 1094). Figure 20.4: Electrochemical Reactions on the Particle Surface (p. 889) schematically shows the electrochemical reactions that take place on the particle surface in the anode during battery charging.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Figure 20.4: Electrochemical Reactions on the Particle Surface



The following electrochemical reactions are considered in the physics-based battery life model:

- Intercalation reaction:

$$C_6 + Li^+ + e^- \leftrightarrow LiC_6 \tag{20.48}$$

- SEI formation reaction:

$$2C_2H_4CO_3 + 2Li^+ + 2e^- \rightarrow (CH_2OCO_2Li)_2 \downarrow + C_2H_4 \uparrow \tag{20.49}$$

- Lithium deposition reaction:

$$Li^+ + e^- \rightarrow Li(s) \downarrow \tag{20.50}$$

The reactions occur when the Li^+ ions are deposited onto the graphite surface and are partly converted to metallic lithium (Li plating) and partly to SEI.

The SEI formation and lithium deposition reactions are unwanted side reactions that compete with the intercalation reaction. These side reactions are only prominent and modeled in the anode during battery charging.

The SEI growth and Li plating cause the battery to age mainly via the following effects:

- Some of the lithium inventory is lost to form the SEI and plated lithium, which leads to a drop in the battery theoretical capacity.
- The surface film of the SEI and plated lithium increases the surface resistance of the anode graphite particles.
- The SEI and plated lithium reduce the porosity of the anode porous media, which further reduces the effective conductivity and diffusivity of the electrolyte.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

To model the SEI growth and lithium plating, the physics-based battery life model solves the evolution of the SEI concentration c_{SEI} using the following formulation:

$$\frac{dc_{SEI}}{dt} = -\frac{a_s}{2F}\left(j^{SEI} + \beta j^{Li}\right) \tag{20.51}$$

and the plated lithium concentration c_{Li} using the following formulation:

$$\frac{dc_{Li}}{dt} = -\frac{a_s}{F}\left(1-\beta\right) j^{Li} \tag{20.52}$$

In Equation 20.51 (p. 890) and Equation 20.52 (p. 890), the following notations are used:

a_s = volumetric solid/electrolyte interfacial area

F = Faraday constant

j^{SEI} = current flux at the active material pore-wall due to SEI formation reaction

j^{Li} = current flux at the active material pore-wall due to lithium deposition reaction

β = fraction of deposited lithium that becomes SEI

As the governing equations of SEI and plated lithium have no spatial term, their evolutions are solely dependent on the local source terms, that is, the local current fluxes of the side reactions. Those source terms are formulated by the cathodic Tafel expressions:

$$j^{SEI} = -Fk_{SEI}c_{s,EC}\exp\left(-\frac{a_{c,SEI}F}{RT}\eta_{SEI}\right) \tag{20.53}$$

and

$$j^{Li} = -i_{Li}\exp\left(-\frac{a_{c,Li}F}{RT}\eta_{Li}\right) \tag{20.54}$$

In the above equations:

k_{SEI} = reaction rate constant

$c_{s,EC}$ = surface concentration of the ethylene carbonate (EC)

$\alpha_{c,SEI}$ and $\alpha_{c,Li}$ = cathodic charge transfer coefficients for the SEI and plated lithium side reactions, respectively

R = universal gas constant

T = battery temperature

η_{SEI} and η_{Li} = overpotentials for the SEI and plated lithium side reactions, respectively

i_{Li} = reaction exchange current density, which is modeled as a constant

The surface concentration of EC, $c_{s,EC}$, is determined by the diffusive flux of EC across the surface film:

$$-D_{EC}\frac{c_{s,EC} - c_{EC}}{\delta_{film}} = -\frac{j^{SEI}}{F} \tag{20.55}$$

where

c_{EC} = concentration of EC in the bulk electrolyte

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

D_{EC} = diffusivity of EC

δ_{film} = film thickness

Similar to Equation 20.19 (p. 879), the overpotential of a side reaction (SEI formation and lithium deposition) η_{side} is computed as:

$$\eta_{side}=\varphi_s-\varphi_e-U_{side}-\left(j^{int}+\sum j^{side}\right)R_{film} \tag{20.56}$$

where

φ_s and φ_e = potentials of the solid (electrode) and electrolyte phases, respectively

j^{int} and j^{side} = active material pore-wall current fluxes due to lithium intercalation reaction and to side reactions, respectively

U_{side} = equilibrium potential of a side reaction

R_{film} = electric resistance

The physical properties of the surface film (thickness δ_{film} and electric resistance R_{film}) are expressed as functions of the SEI and plated lithium concentrations:

$$\delta_{film}=\left(\left(1+\frac{1}{\varepsilon_{s,0}}\left(\frac{M_{SEI}}{\rho_{SEI}}c_{SEI}+\frac{M_{Li}}{\rho_{Li}}c_{Li}\right)\right)^{\frac{1}{3}}-1\right)r_s \tag{20.57}$$

$$R_{film}=\left(\frac{\frac{M_{SEI}}{\rho_{SEI}}c_{SEI}}{\frac{M_{SEI}}{\rho_{SEI}}c_{SEI}+\frac{M_{Li}}{\rho_{Li}}c_{Li}}\right)\frac{\delta_{film}}{\kappa_{SEI}} \tag{20.58}$$

The anode electrolyte porosity ε_e decreases as the surface film grows and is computed as:

$$\varepsilon_e=\varepsilon_{e,0}-\left(\frac{M_{SEI}}{\rho_{SEI}}c_{SEI}+\frac{M_{Li}}{\rho_{Li}}c_{Li}\right) \tag{20.59}$$

In Equation 20.57 (p. 891) through Equation 20.59 (p. 891), the following notations are used:

M_{SEI} and M_{Li} = molecular weights of SEI and plated lithium, respectively

r_s = particle radius of the anode active material (same as that used in Equation 20.25 (p. 879))

κ_{SEI} = ionic conductivity of SEI

$\varepsilon_{e,0}$ = volume fraction of the electrolyte material ε_e of a fresh battery (see Equation 20.21 (p. 879))

$\varepsilon_{s,0}$ = volume fraction of the active material ε_s of a fresh battery (see Equation 20.24 (p. 879))

ρ_{SEI} and ρ_{Li} = densities of SEI and plated lithium, respectively

In practical Newman's P2D model simulations of battery aging through electric load cycles, the physics-based battery life model is solved on the electrode scale in the standalone mode (see Simulating the Aging Process of a Battery (Standalone Mode) in the *Fluent User's Guide* for more information). Further, the standalone simulation results can be included in a 3D CFD simulation to account for the battery aging effects. In this scenario, the aging effects are treated as being frozen in time. More specifically, during the E-chemistry model calculations, the distributions of the SEI and the plated

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

lithium concentrations (which determines the lithium inventory loss), the surface film resistance, and the electrolyte porosity across the anode are included into the Newman's P2D model as steady-state profiles. In a practical CFD simulation constrained by the computing power, the simulation time is much smaller than the time scale of battery life. This wide separation of time scale justifies the frozen-in-time aging effects. See Inputs for the Newman's P2D Model in the *Fluent User's Guide* for details.

20.9. Battery Swelling Model

There are multiple mechanisms that cause swelling and deformation of lithium-ion batteries (LIB), including the change in the volume of electrode particles induced by the lithium insertion/extraction, the growth of the particle surface film due to side reactions, and the electrode elastic deformation due to external mechanical load. The swelling and deformation may change both the electrochemical characteristics and the structural mechanical characteristics (such as, geometry dimensions or internal stress) of the battery. To consider the swelling effect, Ansys Fluent provides an empirical swelling model for the NTGK/DCIR and ECM methods and for the Newman's P2D method with the physics-based model.

For more information about using the battery swelling model, see Setting a Battery Swelling Case in the *Fluent User's Guide*.

20.9.1. Empirical-Based Swelling Model

In the NTGK/DCIR and the ECM models, details of lithium ion transport are not resolved. Instead, the battery's electric behavior is modeled by an electric circuit, and SOC is used to track the battery's charging status. In the empirical swelling model, the swelling strain ϵ is assumed to be a linear function of the battery's SOC level:

$$\epsilon = \Omega\left(soc - soc_{ref}\right) \Big/ \left(1 + \Omega^{*} soc_{ref}\right) \tag{20.60}$$

where Ω is the swelling coefficient that describes how fast the battery swells as its charging status changes, soc is the battery's current SOC level, and soc_{ref} is the reference SOC level of the battery at the swell-neutral state.

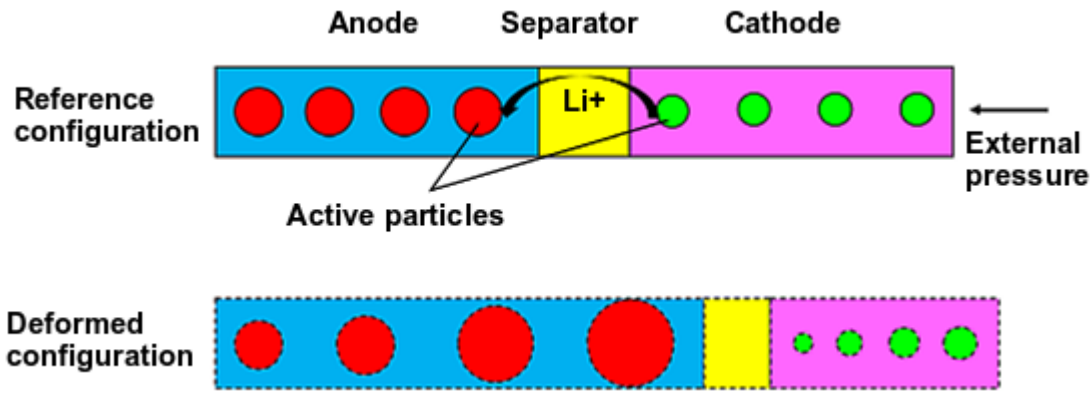
You can customize the swelling strain formulation using a user-defined function (UDF) provided by Ansys Fluent. See `DEFINE_BATTERY_SWELL_STRAIN` in the *Fluent Customization Manual* for details.

20.9.2. Physics-Based Swelling Model

The physics-based battery swelling model is based on the Newman's P2D model (see Newman's P2D Model (p. 877)) and includes the swelling effects. The Newman's P2D model resolves the lithium transport and the lithium insertion/desertion reactions at the scale of the anode-separator-cathode sandwich layer. The swelling effects are modeled by considering the deformation of the electrode active particles during the lithium insertion/desertion reactions and the elastic deformation of the sandwich layer under externally applied pressure. Consequently, during battery charging, the geometric configuration of the P2D model deviates from the initial (or reference) geometric configuration due to deformation as shown schematically in Figure 20.5: Physics-Based Battery Swelling Model: Reference and Deformed Configurations (p. 893).

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Figure 20.5: Physics-Based Battery Swelling Model: Reference and Deformed Configurations



In the swelling battery model, the P2D equations are solved in the reference configuration. For this purpose, the transformation laws based on the continuum mechanics theory are applied to account for the deformation effects. The transformation laws are originally formulated using high-order tensors with the assumptions that the deformation is one-dimensional in the electrode layer and isotropic in the particles. All terms related to deformation degenerate from tensors into scalars. The resulting enhanced P2D model equations are formulated as follows:

- Lithium conservation in the solid phase:

$$\frac{\partial C_s}{\partial t} = \frac{D_s}{r^2}\frac{\partial}{\partial r}\left(r^2 F_s^{-T} F_s^{-1}\frac{\partial C_s}{\partial r}\right) \tag{20.61}$$

- Lithium conservation in the electrolyte phase:

$$\frac{\partial\left(\varepsilon^{neu} c_e\right)}{\partial t} = \frac{\partial}{\partial x}\left(JF^{-1}F^{-T}D_e^{eff}\frac{\partial c_e}{\partial x}\right) + \frac{a_s^{neu}\left(1-t_+^0\right)}{F} J_s F_s^{-1}\left(j^{int} + \sum j^{side}\right) \tag{20.62}$$

- Charge conservation in solid phase:

$$\frac{\partial}{\partial x}\left(JF^{-1}F^{-T}\sigma^{eff}\frac{\partial \varphi_s}{\partial x}\right) - a_s^{neu} J_s F_s^{-1}\left(j^{int} + \sum j^{side}\right) = 0 \tag{20.63}$$

- Charge conservation in the electrolyte phase:

$$\frac{\partial}{\partial x}\left(JF^{-1}F^{-T}\kappa^{eff}\frac{\partial \varphi_s}{\partial x}\right) + \frac{\partial}{\partial x}\left(JF^{-1}F^{-T}\kappa_d^{eff}\frac{\partial \ln\left(c_e\right)}{\partial x}\right) + a_s^{neu} J_s F_s^{-1}\left(j^{int} + \sum j^{side}\right) = 0 \tag{20.64}$$

where

F_s^{-1} = scalar reduced from the deformation gradient inverse of the P2D particle domain

F_s^{-T} = scalar reduced from the deformation gradient inverse-transpose of the P2D particle domain

J_s = determinant of the deformation gradient of the P2D particle domain

F^{-1} = scalar reduced from the deformation gradient inverse of the P2D electrode domain

F^{-T} = scalar reduced from the deformation gradient inverse-transpose of the P2D electrode domain

J = determinant of the deformation gradient of the P2D electrode domain

The remaining nomenclature is the same as in Newman's P2D Model (p. 877).

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

A more comprehensive description of the swelling model theory can be found in [727] (p. 1095).

With deformation taken into account via the transformation terms, the enhanced P2D model equations are solved in the reference domain (that is, using the original P2D mesh). The solid-state lithium concentration in the above model equations represents the value in the reference configuration. It is transformed to the value of the deformed configuration (that is, its true value) when used to calculate any battery properties, such as the transfer current in the Butler-Volmer equation (Equation 20.18 (p. 879) - Equation 20.20 (p. 879)). The transformation of the solid phase lithium concentration from the reference domain into the deformed domain is expressed as:

$$c_s = C_s / J_s \tag{20.65}$$

where

C_s = reference domain value

c_s = deformed domain value

Both the particle lithium insertion and external mechanical load contribute to the domain deformations, while the effects of particle surface film growth are not included. The following transformation terms are used to account for the deformation effects:

$$\bar{\Omega}_s = \omega\left(\frac{\bar{C}_s}{C_{s,max}} - \frac{C_s^{neu}}{C_{s,max}}\right) \tag{20.66}$$

$$J_s = 1 + \bar{\Omega}_s \tag{20.67}$$

$$F_s^{-1} = F_s^{-T} = J_s^{-1/3} \tag{20.68}$$

$$J = 1 + \varepsilon_s^{neu}\bar{\Omega}_s + \frac{\sigma_{nn}}{E} \tag{20.69}$$

$$F^{-1} = F^{-T} = J^{-1} \tag{20.70}$$

Equation 20.66 (p. 894) quantifies the coefficient of particle domain volume change $\bar{\Omega}_s$, which varies linearly depending on the state of lithiation (that is, the lithium concentration averaged over the particle domain $\bar{C}_s$ normalized by the theoretical maximum concentration of lithium $C_{s,max}$) of the electrode active material. Here, ω is the swelling coefficient that describes how fast the particles swell during lithium insertion, and C_s^{neu} is the lithium concentration in the swell-neutral state, which acts as an offset in a linear formulation that marks the swell-neutral state, that is, where there is no change in the particle volume. Both ω and C_s^{neu} are user-specified model parameters.

In Equation 20.69 (p. 894), ε_s^{neu} is the active material porosity at the swell-neutral state, σ_{nn} is the mechanical pressure exerted normal to the battery electrode sandwich layer, and E is Young's modulus. The second and third terms in Equation 20.69 (p. 894) are the electrode deformation induced by the particles swelling and the electrode elastic deformation, respectively. ε_s^{neu} and E are user-specified parameters. See Coupling between the Swelling and FSI Structural Models (p. 895) for more details about σ_{nn}.

In addition to deformation, the model formulation also includes a change in the electrode porosity. In Equation 20.61 (p. 893)-Equation 20.64 (p. 893), with the exception of some reference properties (marked with the superscript neu), all the other porosity-dependent properties (for example, the effective conductivity κ^{eff} and diffusivity D_e^{eff} in Equation 20.21 (p. 879) and Equation 20.22 (p. 879)) are obtained with the following modified porosities that include the swelling effects:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

- Electrolyte porosity:

$$\varepsilon_e = \frac{1}{J}\varepsilon_e^{neu} \tag{20.71}$$

- Active material porosity:

$$\varepsilon_s = \frac{J_s}{J}\varepsilon_s^{neu} \tag{20.72}$$

As in the case of the swell-neutral state lithium concentration in Equation 20.66 (p. 894), the swell-neutral state porosities in Equation 20.71 (p. 895) and Equation 20.72 (p. 895) are also user-specified model parameters.

Note:

The porosity of the filler material is implicitly affected by the swelling effects, since the sum of the porosities of all different material components should be equal to unity.

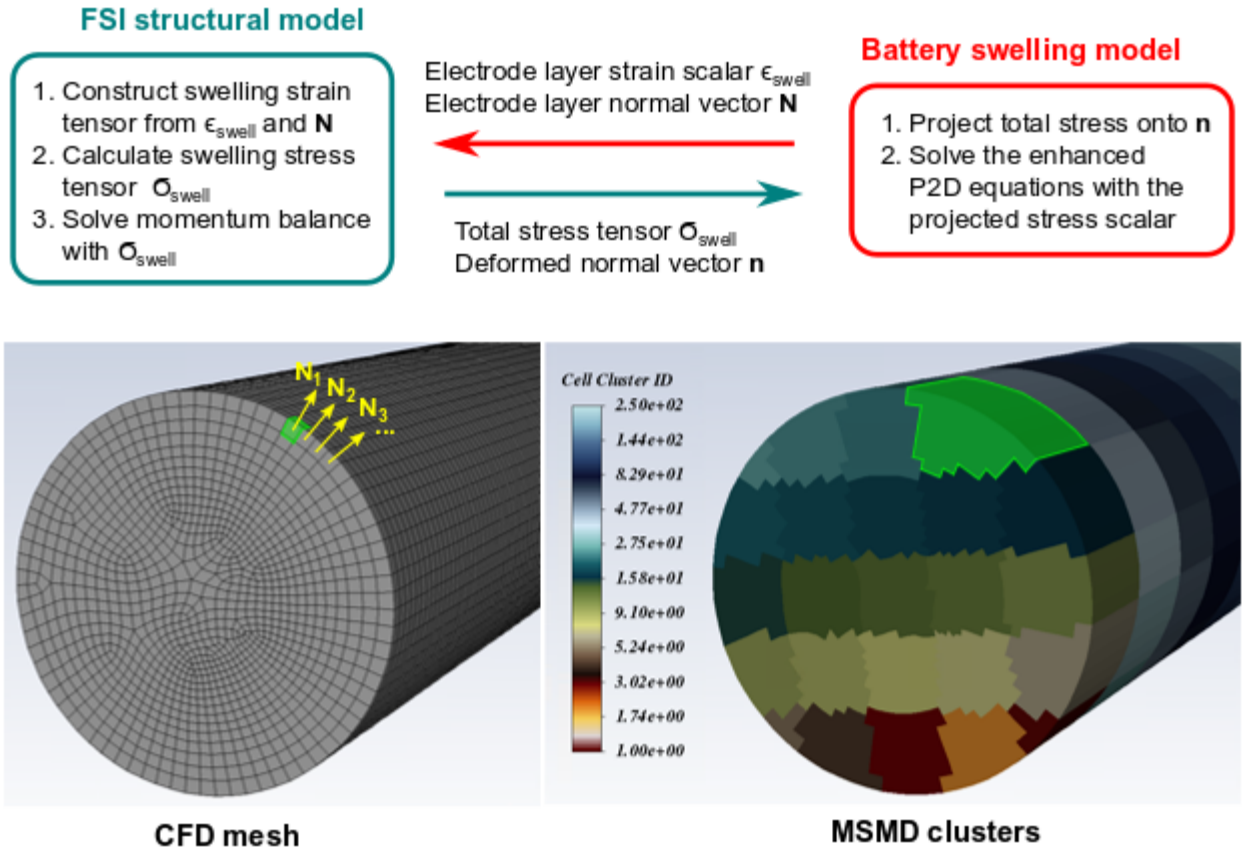
20.9.3. Coupling between the Swelling and FSI Structural Models

When the MSMD battery model is coupled with the fluid-structure interaction (FSI) structural model, the electrode layer deformation is provided by the battery submodel, while the macroscale battery cell deformation is solved by the FSI structural model. The model coupling methods are adapted from [726] (p. 1095) and are shown in Figure 20.6: Coupling of the Swelling and FSI Structural Models (p. 896).

For the Newman's P2D model, the coupling between the structure and battery models is two-way. This kind of coupling is more comprehensive as further described.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Figure 20.6: Coupling of the Swelling and FSI Structural Models



During the simulation, the battery swelling model calculates the effective strain induced by the swelling of the battery sandwich layer embedded within the CFD control volume and passes it to the structural model. In its turn, the structural model calculates the effective external pressure exerted on the battery sandwich layer and passes it to the battery swelling model. In addition, since the battery swelling model is one-dimensional while the FSI structural model is three-dimensional, a directional information (that is, the normal vector of the electrode layer) is passed between the two models to convert the coupling variables back and forth between scalars and tensors.

Figure 20.6: Coupling of the Swelling and FSI Structural Models (p. 896) illustrates two different levels of spatial resolutions of the coupling variables. The FSI structural model is solved on the CFD mesh, while the Echem submodel is solved iteratively over MSMD cell clusters, which are CFD control volumes grouped together to save computational cost. A single CFD control volume and its corresponding MSMD cell cluster are shown in green side-by-side. When the coupling variables are passed from the battery to the structure solver, the single cluster value of the swelling strain scalar is assigned to all CFD control volumes within the cluster. When the coupling variables are passed from the structure to the battery solver, the projected stress scalar is volume-averaged among all CFD control volumes within the cluster.

When solving the E-Chem submodel, the mechanical load input of the mechanical pressure σ_{nn} (used in Equation 20.69 (p. 894)) is calculated as the normal component of the total stress tensor projected onto the local electrode layer surface:

$$\sigma_{nn}=\mathbf{n}^{\mathrm{T}}\sigma_{t}\mathbf{n} \tag{20.73}$$

where

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

σ_t = total stress tensor obtained from the FSI structural model

$\mathbf{n}$ = normal vector of the battery electrode layer in the deformed coordinate system, which is calculated based on the user-specified normal vector $\mathbf{N}$ of the reference frame and includes the effects of geometry deformation

The strain scalar of the electrode layer ϵ_{swell} is calculated as the swell-induced deformation averaged over the P2D electrode domain:

$$\epsilon_{swell}=\frac{1}{l_n+l_s+l_p}\int_{x=0}^{x=l_n+l_s+l_p}\varepsilon_s^{neu}\bar{\Omega}_s\,dx \tag{20.74}$$

where l_n, l_p, and l_s are the swell-neutral thickness of the negative electrode, positive electrode, and separator, respectively. The remaining nomenclature is the same as in Equation 20.69 (p. 894).

The enhanced P2D model is then solved using Equation 20.61 (p. 893) through Equation 20.64 (p. 893). The strain scalar ϵ_{swell} and the reference frame normal vector $\mathbf{N}$ are then passed to the FSI structural model. The FSI structural model solves the battery geometry deformation and passes the resulting total stress σ_t and the deformed frame normal vector $\mathbf{n}$ back to the Echem submodel.

For the NTGK/DCIR and ECM models, the coupling between the structure and battery solvers is one-way. Just like in the case with the physics-based swelling model, the swelling strain ε and the reference fame normal vector $\mathbf{N}$ are passed to the structure solver. However, the structure solver does not pass anything back to the battery solver.

Note:

The solutions of the FSI structural model satisfy the solid mechanics momentum balance with swelling stress included and are affected by the solid materials properties (such as strain-stress relation and boundary constraints). For more information, see The Structural Model for Intrinsic Fluid-Structure Interaction (FSI) (p. 827). For details about coupling formulations and vector transformation in the FSI structural model, see Coupling of the Structural Model with the Battery Swelling Model (p. 837).

20.9.4. Modeling Swelling in the E-Chem Standalone Mode

Similar to the default Newman's P2D model, the physics-based battery swelling model can also be solved in the E-Chem standalone mode. In this case, the following types of mechanical load can be modeled for the battery sandwich layer in Ansys Fluent:

- External pressure exerted normal to the electrode layer P_{ext}

  The effective mechanical pressure σ_{nn} is obtained as:

  $$\sigma_{nn}=-P_{ext} \tag{20.75}$$

- Total thickness change of the electrode layer ΔL_{tot}

  The effective mechanical pressure σ_{nn} is inversely calculated by the following equation to meet the target total thickness change:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$\frac{1}{l_n+l_s+l_p}\int_{x=0}^{x=l_n+l_s+l_p}\left(\varepsilon_s^{neu}\bar{\Omega}_s+\frac{\sigma_{nn}}{E}\right)dx=\frac{\Delta L_{tot}}{l_n+l_s+l_p} \tag{20.76}$$

- Effective stiffness of the electrode layer mounting E_{ext} (for example, the battery case)

  The effective mechanical pressure σ_{nn} is inversely calculated by the following equation to meet the stress-strain-relation under external stiffness:

$$\frac{1}{l_n+l_s+l_p}\int_{x=0}^{x=l_n+l_s+l_p}\left(\varepsilon_s^{neu}\bar{\Omega}_s+\frac{\sigma_{nn}}{E}\right)dx=-\frac{\sigma_{nn}}{E_{ext}} \tag{20.77}$$

In Equation 20.76 (p. 898) and Equation 20.77 (p. 898), the nomenclature used is the same as in Equation 20.69 (p. 894) and Equation 20.74 (p. 897). The left-hand side represents the total electrode deformation strain averaged over the sandwich layer, and the right-hand side represents the effective load strain.

Refer to Running the Standalone Echem Model in the *Fluent User's Guide* for more information.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Chapter 21: Modeling Fuel Cells

The Ansys Fluent Fuel Cell Modules Chapter provides a theoretical discussion of the models used in Ansys Fluent. The available Ansys Fluent add-on fuel cell modules are:

- PEMFC Model - allows you to model polymer electrolyte membrane fuel cells (PEMFC) with (or without) micro-porous layers. This model is a new and recommended model for simulating energy conversion processes in a PEM fuel cell. For more information, see Using the PEMFC Model.
- Fuel Cell and Electrolysis Model - allows you to model polymer electrolyte membrane fuel cells (PEMFC), solid oxide fuel cells (SOFC), and electrolysis with Ansys Fluent. This model is sometimes referred to as the Resolved Electrolyte model. Note that the PEMFC sub-model can be used to model both the low-temperature and high-temperature PEM fuel cells. For more information, see Using the Fuel Cell and Electrolysis Model.
- SOFC With Unresolved Electrolyte Model - allows you to model solid oxide fuel cells (SOFC). For more information, see Using the Solid Oxide Fuel Cell With Unresolved Electrolyte Model.

This chapter is divided into the following sections:

21.1. PEMFC Model Theory

21.2. Fuel Cell and Electrolysis Model Theory

21.3. SOFC Fuel Cell With Unresolved Electrolyte Model Theory

21.1. PEMFC Model Theory

This chapter presents the theoretical background for the advanced Polymer Electrolyte Membrane Fuel Cell (PEMFC) modeling capabilities in Ansys Fluent.

21.1.1. Introduction

21.1.2. Electrochemistry Modeling

21.1.3. Current and Mass Conservation

21.1.4. Water Transport and Mass Transfer in PEMFC

21.1.5. Heat Source

21.1.6. Properties

21.1.7. Transient Simulations

21.1.8. Leakage Current (Cross-Over Current)

21.1.9. Zones Where User-Defined Scalars Have Physically Meaningful Values

21.1.1. Introduction

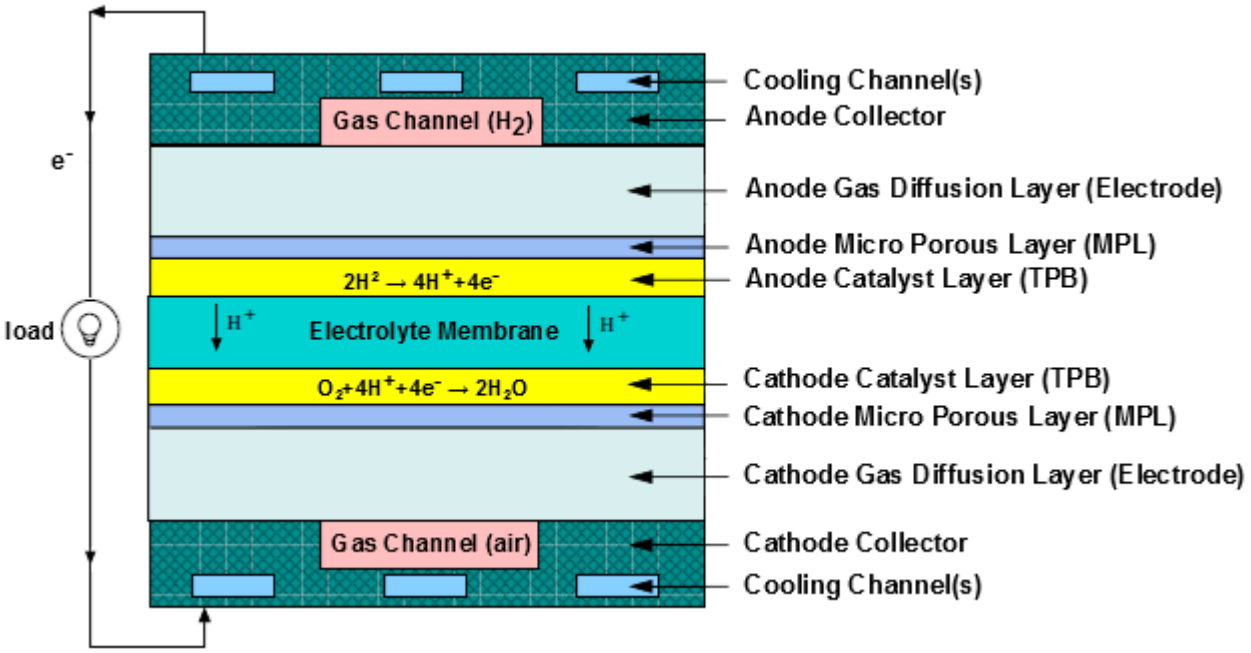
The PEMFC module is provided as an add-on module with the standard Ansys Fluent licensed software.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

A fuel cell is an energy conversion device that converts the chemical energy of fuel into electrical energy. With the PEMFC model, both the Triple Phase Boundary (TPB), also known as the catalyst layer, and the ionic conducting electrolyte, also known as the membrane in PEMFC terminology, are included in the computational domain. The PEMFC module allows you to model polymer electrolyte membrane fuel cells.

To determine the physical domains that are included in the PEMFC module, a schematic of a polymer electrolyte membrane fuel cell (PEMFC) is shown in Figure 21.1: Schematic of a PEM Fuel Cell (p. 900).

Figure 21.1: Schematic of a PEM Fuel Cell



Hydrogen flows into the fuel cell on the anode side. It diffuses through the porous gas diffusion layer (GDL) and micro-porous layer (MPL), which is optional for the operation of a PEMFC, and then comes in contact with the catalyst layer. Here it forms hydrogen ions and electrons. The hydrogen ions diffuse through the polymer electrolyte membrane at the center, the electrons flow through the gas diffusion layer to the current collectors and into the electric load attached. Electrons enter the cathode side through the current collectors and the gas diffusion layer. Similarly, oxygen (or air) flows into the fuel cell on the cathode side and diffuses through the porous gas diffusion layer and then micro-porous layer to reach the catalyst layer. At the catalyst layer, the electrons, the hydrogen ions, and the oxygen combine to form water.

In the PEMFC model in Ansys Fluent, two electric potential fields are solved. One potential is solved in the membrane and the catalyst layer. The other is solved in the TPB catalyst layer, the micro-porous layer, the porous electrode, and the current collectors. The rates of electrochemical reactions are computed in the TPB layers at both the anode and the cathode. Based on the cell voltage that you prescribe, the current density value is computed. Alternatively, a cell voltage can be computed based on a prescribed average current density.

The polymer electrolyte membrane fuel cell (PEMFC) has emerged as a favored technology for auto transportation and power generation because it is compact, clean, runs at low temperature (<100° C), permits an adjustable power output, and can be started relatively rapidly. Hydrogen is supplied

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

at the anode and air is supplied at the cathode. The following electrochemical reactions take place in the anode and cathode triple phase boundary (TPB) layers, respectively,

$$H_2 \Leftrightarrow 2H^+ + 2e^- \quad (anode\ TPB) \tag{21.1}$$

$$\frac{1}{2}O_2 + 2e^- + 2H^+ \Leftrightarrow H_2O \quad (cathode\ TPB) \tag{21.2}$$

Electrons produced in the anode travel through an external circuit to the cathode, while protons (H^+) travel through the membrane from the anode TPB to the cathode TPB, thereby forming an electrical circuit.

In a PEM fuel cell, the three phases of water are present. The gas and liquid water phases are present in all the physical domains except the solid membrane and current collectors. Water is also present in the dissolved phase, but only inside the catalyst layers and the membrane. The water produced by the cathode side electro-chemistry is assumed to be in the dissolved phase (Equation 21.2 (p. 901)).

21.1.2. Electrochemistry Modeling

At the center of the electrochemistry is the computation of the rates of the anodic and cathodic reactions. The electrochemistry model adopted in Ansys Fluent is one that has been used by other groups ([323] (p. 1071), [416] (p. 1077), and [660] (p. 1091)).

The driving force behind these reactions is the surface overpotential: the difference between the phase potential of the solid and the phase potential of the electrolyte/membrane. Therefore, two potential equations are solved. One potential equation (Equation 21.3 (p. 901)) accounts for the electron transport of e^- through the solid conductive materials and is solved in the TPB catalyst layer, the solid grids of the porous media, and the current collector; the other potential equation (Equation 21.4 (p. 901)) represents the protonic (that is, ionic) transport of H^+ and is solved in the TPB catalyst layer and the membrane. The two potential equations are as follows:

$$\nabla \cdot \left(\sigma_{sol} \nabla \phi_{sol} \right) + R_{sol} = 0 \tag{21.3}$$

$$\nabla \cdot \left(\sigma_{mem} \nabla \phi_{mem} \right) + R_{mem} = 0 \tag{21.4}$$

where

σ = electrical conductivity (1/ohm-m)

ϕ = electric potential (volts)

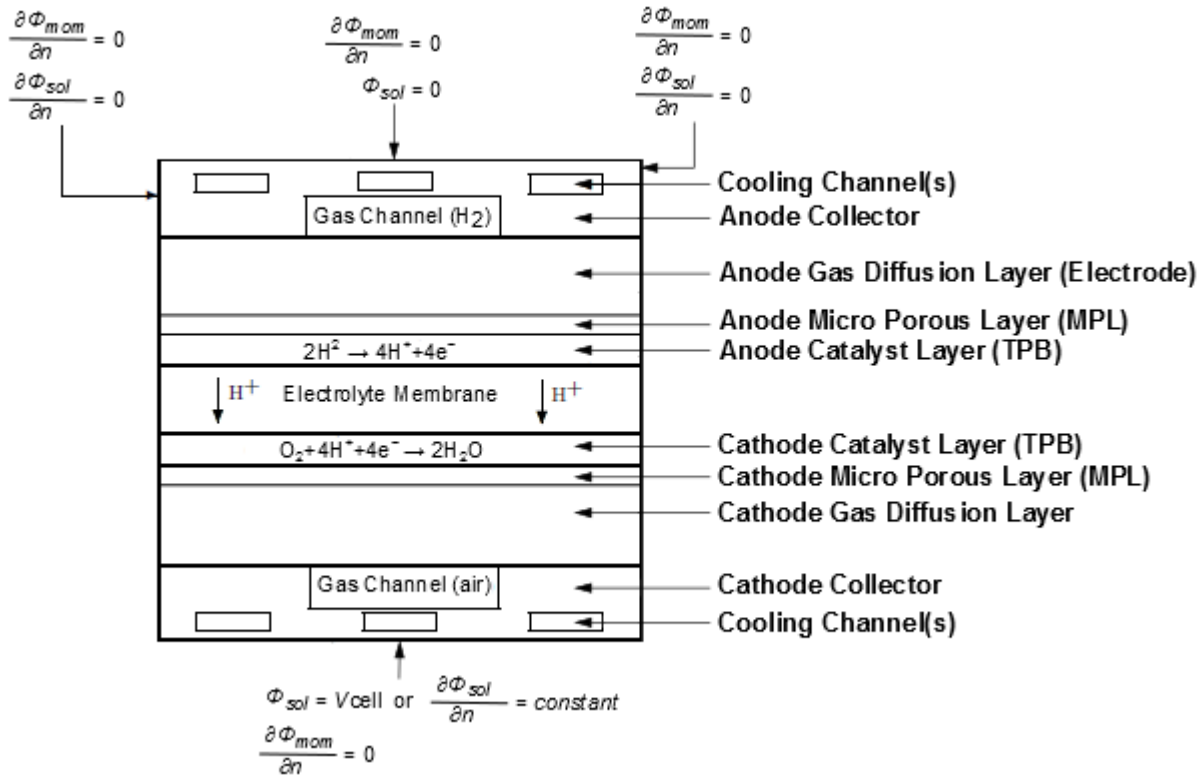
R = volumetric transfer current (A/m^3)

In the above equations, subscripts mem and sol refer to the membrane and solid phases, respectively.

The following figure illustrates the boundary conditions that are used to solve for ϕ_{sol} and ϕ_{mem}.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Figure 21.2: Boundary Conditions for the Electric Potentials (Solid and Membrane) — PEM Fuel Cell



There are two types of external boundaries: those that have an electrical current passing through them, and those that do not.

As no ionic current leaves the fuel cell through any external boundary, there is a zero flux boundary condition for the membrane phase potential, ϕ_{mem}, on all outside boundaries.

For the solid phase potential, ϕ_{sol}, there are external boundaries on the anode and the cathode side that are in contact with the external electric circuit. Electrical current generated in the fuel cell only passes through these boundaries. On all other external boundaries there is a zero flux boundary condition for ϕ_{sol}.

On the external contact boundaries, fixed values for ϕ_{sol} (potentiostatic boundary conditions) are recommend. If the anode side is set to zero, the (positive) value prescribed on the cathode side is the cell voltage. Specifying a constant flux (say on the cathode side) means to specify galvanostatic boundary conditions.

The transfer currents, or the source terms in Equation 21.3 (p. 901) and Equation 21.4 (p. 901), are nonzero only inside the catalyst layers and are computed as:

- For the potential equation in the solid phase, $R_{sol}=-R_{an}\left(<0\right)$ on the anode side and $R_{sol}=+R_{cat}\left(>0\right)$ on the cathode side.
- For the potential equation in the membrane phase, $R_{mem}=+R_{an}\left(>0\right)$ on the anode side and $R_{mem}=-R_{cat}\left(<0\right)$ on the cathode side.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

The source terms in Equation 21.3 (p. 901) and Equation 21.4 (p. 901), also called the exchange current density (A/m^3), have the following general definitions:

$$R_{an}=\left(\zeta_{an} j_{an}(T)\right)\left(\frac{[A]}{[A]_{ref}}\right)^{\gamma_{an}}\left(e^{\alpha_{an}^{an} F\eta_{an}/RT}-e^{-\alpha_{cat}^{an} F\eta_{an}/RT}\right) \tag{21.5}$$

$$R_{cat}=\left(\zeta_{cat} j_{cat}(T)\right)\left(\frac{[C]}{[C]_{ref}}\right)^{\gamma_{cat}}\left(-e^{\alpha_{an}^{cat} F\eta_{cat}/RT}+e^{-\alpha_{cat}^{cat} F\eta_{cat}/RT}\right) \tag{21.6}$$

where

$j(T)$ = reference exchange current density per active surface area (A/m^2)

ζ = specific active surface area (1/m)

$[\,]$, $[\,]_{ref}$ = local species concentration, reference value ($\mathrm{kmol/m^3}$)

γ = concentration dependence

α_{an}^{an} and α_{cat}^{an} = anode and cathode transfer coefficients of the anode electrode, respectively (dimensionless)

α_{an}^{cat} and α_{cat}^{cat} = anode and cathode transfer coefficients of the cathode electrode, respectively (dimensionless)

η_{an} = surface overpotential given by Equation 21.11 (p. 904)

η_{cat} = surface overpotential given by Equation 21.12 (p. 904)

F = Faraday constant (9.65×10^7 C/kmol)

R = the universal gas constant

T = temperature

The above equation is the general formulation of the Butler-Volmer function. Note that the effects of the number of electrons in electrochemistry reactions are accounted for in the transfer coefficients. A simplification to this is the Tafel formulation given by:

$$R_{an} = \zeta_{an} j_{an}(T)\left(\frac{[A]}{[A]_{ref}}\right)^{\gamma_{an}}\left(e^{\alpha_{an} F\eta_{an}/RT}\right) \tag{21.7}$$

$$R_{cat} = \zeta_{cat} j_{cat}(T)\left(\frac{[C]}{[C]_{ref}}\right)^{\gamma_{cat}}\left(e^{-\alpha_{cat} F\eta_{cat}/RT}\right) \tag{21.8}$$

By default, the Butler-Volmer function is used in the Ansys Fluent PEMFC model to compute the transfer currents inside the catalyst layers. When the magnitude of the surface over-potential (η) is large, the Butler-Volmer formulation reduces to the Tafel formulation.

In Equation 21.5 (p. 903) through Equation 21.8 (p. 903), $[A]$ and $[C]$ represent the molar concentration of the species upon which the anode and cathode reaction rates depend, respectively. That is, A represents H_2 and C represents O_2.

The reference exchange current density $j_{an}(T)$ and $j_{cat}(T)$ are dependent on the local temperature as follows:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$j_{an}(T) = j_{an}^{ref} e^{-E_{an}/RT\left(1-T/T_{an}^{ref}\right)} \tag{21.9}$$

$$j_{cat}(T) = j_{cat}^{ref} e^{-E_{cat}/RT\left(1-T/T_{cat}^{ref}\right)} \tag{21.10}$$

where

E_{an} and E_{cat} = user-specified activation energy

T_{ref} = user-specified reference temperature

j_{an}^{ref} and j_{cat}^{ref} = reference exchange current density at a specified reference temperature

The driving force for the kinetics is the local surface overpotential, η, also known as the *activation loss*. It is generally the difference between the solid and membrane potentials, ϕ_{sol} and ϕ_{mem}.

$$\eta_{an} = \phi_{sol} - \phi_{mem} - U_{an}^{0} \tag{21.11}$$

$$\eta_{cat} = \phi_{sol} - \phi_{mem} - U_{cat}^{0} \tag{21.12}$$

The half cell potentials at anode and cathode U_{an}^{0} and U_{cat}^{0} are computed by Nernst equations as follows [583] (p. 1086):

$$U_{an}^{0} = E_{an}^{0} - \frac{\Delta S_{an}}{2F}\left(T - T^{0}\right) - \frac{RT}{2F}\ln\left(\frac{p_{H_2}}{p^{0}}\right) \tag{21.13}$$

$$U_{cat}^{0} = E_{cat}^{0} + \frac{\Delta S_{cat}}{2F}\left(T - T^{0}\right) - \frac{RT}{2F}\ln\left(\frac{p_{H_2O}}{p_{sat}\sqrt{p_{O_2}/p^{0}}}\right) \tag{21.14}$$

where p_{sat} is the water saturation pressure (Equation 21.50 (p. 913)) and p_{H_2}, p_{O_2}, and p_{H_2O} are the partial pressures of hydrogen, oxygen, and water vapor, respectively. In the above equations, the standard state (T^{0}, p^{0}), the reversible potentials E_{an}^{0} and E_{cat}^{0}, and the reaction entropies ΔS_{an} and ΔS_{cat} are user-specified quantities.

From Equation 21.3 (p. 901) through Equation 21.14 (p. 904), the two potential fields can be obtained.

21.1.2.1. The Cathode Particle Model

When Equation 21.8 (p. 903) is used to compute the cathode transfer current, the mass transport resistance in the catalyst microstructure is not considered ([583] (p. 1086)). The resistance may consist of two parts:

- Resistance due to an ionomer film $\mathfrak{R}_{ion}$
- Resistance due to a liquid water film surrounding particles $\mathfrak{R}_{liq}$

In the Ansys Fluent PEMFC model, including these resistances in calculations of the transfer current is optional. The volumetric transfer current inside the cathode layers is represented by:

$$R_{cat} = 4F\frac{c_{O_2}}{c_{O_2}/j_{O_2}^{ideal} + \mathfrak{R}_{ion} + \mathfrak{R}_{liq}} \tag{21.15}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

where c_{O_2} is the concentration of oxygen at the wall. The $\mathfrak{R}_{ion}$ is a user-specified value, and the $\mathfrak{R}_{liq}$ is calculated by:

$$\mathfrak{R}_{liq} = \frac{\zeta_{cat} r_p^2}{K_w D_w} \cdot \frac{\sqrt[3]{1+\frac{s\varepsilon}{1-\varepsilon}} - 1}{3(1-\varepsilon)} \tag{21.16}$$

where

ζ_{cat} = specific active surface area for the cathode catalyst (1/m)

s = liquid saturation

ε = porosity

r_p = particle radius

$K_w D_w$ = product of oxygen solubility and diffusivity in liquid water (on the order of 10^{-10} m^2/s)

The $j_{O_2}^{ideal}$ in Equation 21.15 (p. 904) is calculated as:

$$j_{O_2}^{ideal} = \frac{R_{cat}^0}{4F} \tag{21.17}$$

Here, R_{cat}^0 is the ideal transfer current computed using Equation 21.6 (p. 903), but without considering resistance.

21.1.3. Current and Mass Conservation

Volumetric source terms (kg/m^3-s) for H_2, O_2, and the dissolved water content λ in the triple-phase boundaries (catalyst layers) due to electrochemical reactions are:

$$S_{H_2} = -\frac{M_{w,H_2}}{2F} R_{an} < 0 \tag{21.18}$$

$$S_{O_2} = -\frac{M_{w,O_2}}{4F} R_{cat} < 0 \tag{21.19}$$

$$S_{\lambda} = \frac{M_{w,H_2O}}{2F} R_{cat} > 0 \tag{21.20}$$

In the above equations, M_{w,H_2O}, M_{w,O_2}, and M_{w,H_2} are the molecular mass of water, oxygen and hydrogen, respectively, F is the Faraday constant, and 2 and 4 are the numbers of electrons per mole of reactants and products.

Since the total electrical current produced in the cathode and the anode catalyst layer, respectively, is the same, we have the following equation for current conservation:

$$\int_{anode} R_{an} dV = \int_{cathode} R_{cat} dV \tag{21.21}$$

The performance and efficiency of the PEM fuel cell may be affected by the nitrogen (N_2) crossover through the membrane. N_2 is transported from the cathode to anode side due to the crossover effect, where the driving force is the concentration gradient across the membrane. The N_2 concentration

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

gradient can be expressed through the N_2 partial pressures in the cathode and anode catalyst layers. The nitrogen flux J_{N_2} is calculated based on the difference of the partial pressures:

$$J_{N_2}=K_{N_2}\frac{P_{N_2,cat}-P_{N_2,an}}{\delta_{mem}} \tag{21.22}$$

where

$P_{N_2,cat}$ and $P_{N_2,an}$ = N_2 partial pressures in the cathode and anode catalyst layers, respectively (Pa)

δ_{mem} = membrane thickness (m)

$K_{N_2}K_{N_2}$ = crossover coefficient (kmol/(s m Pa))

K_{N_2} is calculated based on an empirical formulation [534] (p. 1084):

$$K_{N_2}=\alpha_{N_2}\left(0.0295+1.21f_v-1.93f_v^2\right)\times 10^{-14}\exp\left(\frac{E_{N_2}}{R}\left(\frac{1}{T_{ref}}-\frac{1}{T}\right)\right) \tag{21.23}$$

where

α_{N_2} = N_2 crossover scaling factor

E_{N_2} = activation energy for N_2 equal to 24 kJ/mol

T_{ref} = reference temperature equal to 303 K

f_v = volumetric ratio of water in the membrane given by Equation 21.24 (p. 906)

$$f_v=\frac{\lambda V_w}{V_{mem}+\lambda V_w} \tag{21.24}$$

where λ is the water content, and V_w and V_{mem} are the molar volumes of the dry membrane and liquid water, respectively.

In the Ansys Fluent PEMFC model, the N_2 crossover flux is treated as a source term in the first layer of mesh cells near the interface between the catalyst layers and membrane.

21.1.4. Water Transport and Mass Transfer in PEMFC

As stated earlier, water is present in three phases in a PEM fuel cell. Depending on local thermodynamic and fluid dynamic conditions, mass transfer may occur between the three phases. For example, since PEM fuel cells operate under relatively low temperature (<100° C), the water vapor may condense to liquid water, especially at high current densities. Dissolved water is generated by cathode-side reactions, and, depending on the local state of deviation from equilibrium, part of it may convert to either the liquid or gas phase. The dissolved phase can also be transported across the membrane from cathode to anode, or from anode to cathode.

While the presence of water keeps the membrane hydrated (which is necessary for PEM fuel cell operation), the liquid water blocks the gas diffusion passage and decreases the diffusion rate and the effective reacting surface area hence reducing the cell performance. Therefore, water formation and transport should be considered when modeling PEMFC systems. In this section, the modeling approaches adopted in Ansys Fluent are described.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

21.1.4.1. The Dissolved Phase Model

The dissolved phase exists in the catalyst layers (ionomers) and the membrane. The generation and transport of dissolved water is described by [715] (p. 1094):

$$\frac{\partial}{\partial t}\left(\varepsilon_i M_{w,H2O}\frac{\rho_i}{EW}\lambda\right)+\nabla\cdot\left(\vec{i}_m\frac{n_d}{F}M_w\right)=\nabla\cdot\left(M_w D^i_w\nabla\lambda\right)+S_\lambda+S_{gd}+S_{ld} \tag{21.25}$$

where

ε_i = porosity of porous media. ε_i = 1 if the membrane is solid.

$\vec{i}_m$ = the ionic current density calculated as $\vec{i}_m=-\sigma_{mem}\nabla\phi_{mem}$.

λ = dissolved water content.

n_d = osmotic drag coefficient.

D^i_w = diffusion coefficient of water content.

S_λ = water generation rate due to cathode side reaction in the catalyst layer (Equation 21.20 (p. 905)).

S_{gd} = rate of mass change between gas and dissolved phases.

S_{ld} = rate of mass change between liquid and dissolved phases.

The S_{gd} and S_{ld} are expressed as ([583] (p. 1086)):

$$S_{gd}=\left(1-s^\theta\right)\gamma_{gd}M_{w,H_2O}\frac{\rho_i}{EW}\left(\lambda_{eq}-\lambda\right) \tag{21.26}$$

$$S_{ld}=s^\theta\gamma_{ld}M_{w,H_2O}\frac{\rho_i}{EW}\left(\lambda_{eq}-\lambda\right) \tag{21.27}$$

where

ρ_i = dry ionomer, or membrane, density (that is, the density of the solid material in porous catalyst layers and the membrane).

EW = equivalent weight of the membrane.

s = liquid saturation.

λ_{eq} = equilibrium water content.

γ_{gd} and γ_{ld} = gas and liquid mass exchange rate constants.

The γ_{gd}, γ_{ld} and θ are user-specified parameters.

The equilibrium water content is computed as ([583] (p. 1086)):

$$\lambda_{eq}=0.3+6a\left(1-\tanh\left(a-0.5\right)\right)+0.69\left(\lambda_{a=1}-3.52\right)a^{0.5}\left(1+\tanh\left(\frac{a-0.89}{0.23}\right)\right)+s\cdot\left(\lambda_{s=1}-\lambda_{a=1}\right) \tag{21.28}$$

where a is the water activity defined as:

$$a=p_{wv}/p_{sat} \tag{21.29}$$

where p_{wv} is the water vapor partial pressure, and p_{sat} is the saturation pressure.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Both $\lambda_{s=1}$ and $\lambda_{a=1}$ in Equation 21.28 (p. 907) are user-specified parameters.

21.1.4.2. The Liquid Phase Model

Liquid water is present in all the porous electrodes and gas channels.

21.1.4.2.1. Liquid Water Transport Equation in the Porous Electrode and the Membrane

The driving force of the liquid water transport is the liquid pressure gradient ∇p_l ([583] (p. 1086)):

$$\frac{\partial}{\partial t}(\varepsilon_i \rho_l s) = \nabla \cdot \left(\frac{\rho_l K K_r}{\mu_l} \nabla p_l \right) + S_{gl} - S_{ld} \tag{21.30}$$

where

ρ_l = liquid water density

μ_l = liquid dynamic viscosity

K = absolute permeability

K_r = relative permeability

p_l = liquid pressure

S_{gl} = rate of mass change between gas and liquid phases

In the porous gas diffusion and micro-porous layers, the relative permeability is computed as:

$$K_r = s^b \tag{21.31}$$

where s is liquid saturation, and b is a user-defined constant.

In the membrane, the relative permeability K_r is expressed as:

$$K_r = \left(\frac{\frac{M_{w,H_2O}}{\rho_l} \lambda_{s=1} + \frac{EW}{\rho_i}}{\frac{M_{w,H_2O}}{\rho_l} \lambda + \frac{EW}{\rho_i}} \cdot \frac{\lambda}{\lambda_{s=1}} \right)^2 \tag{21.32}$$

Replacing p_l in Equation 21.30 (p. 908) with the sum of the capillary pressure p_c and the gas pressure p, Equation 21.30 (p. 908) can be rewritten as:

$$\frac{\partial}{\partial t}(\varepsilon_i \rho_l s) = \nabla \cdot \left(\frac{\rho_l K K_r}{\mu_l} \nabla (p_c + p) \right) + S_{gl} - S_{ld} \tag{21.33}$$

The mass transfer rate between the gas and the liquid phases is computed based on the unidirectional diffusion theory [42] (p. 1055) and [583] (p. 1086):

$$S_{gl} = \begin{cases} \gamma_e \varepsilon s D_{gl} \frac{M_{w,H_2O}}{RT} p \ln\left(\frac{p - p_{sat}}{p - p_{wv}} \right), & p_{wv} <= p_{sat} \\ \gamma_c \varepsilon (1 - s) D_{gl} \frac{M_{w,H_2O}}{RT} p \ln\left(\frac{p - p_{sat}}{p - p_{wv}} \right), & p_{wv} > p_{sat} \end{cases} \tag{21.34}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

where ε is the porosity, γ_e is the evaporation rate coefficient, γ_c is the is the condensation rate coefficient, and D_{gl} has the following form:

$$D_{gl} = \begin{cases} 0.365 \cdot 10^{-4} \left(\frac{T}{343}\right)^{2.334} \cdot \left(\frac{10^5}{p}\right) & cathode \\ 1.79 \cdot 10^{-4} \left(\frac{T}{343}\right)^{2.334} \cdot \left(\frac{10^5}{p}\right) & anode \end{cases}$$

The Ansys Fluent PEMFC module solves for the capillary pressure p_c (Equation 21.33 (p. 908)). Then, because capillary pressure is a function of saturation, liquid saturation can be computed. Note that even though the capillary pressure is continuous across various porous zones, liquid saturation can be discontinuous at the zone interfaces.

Equation 21.33 (p. 908) is solved in regions located in between the anode GDL-channel interface and the cathode GDL-channel interface (gas-diffusion layers, microporous layers, catalytic layers, and membrane). Inside the membrane, the transient term is zero. At GDL-channel interfaces, the liquid water flux is assumed to go out of the GDL and into the gas channel only. No backflow is allowed. The flux is assumed to be driven by the capillary pressure ([583] (p. 1086)) and the dynamic pressure in the gas channel:

$$f_{liq} = \Theta \varepsilon s \cdot \max\left[\left(p_c + \frac{1}{2}\rho V^2\right), 0\right] \tag{21.35}$$

where Θ is the coefficient of liquid water removal, and ρV^2 is the local dynamic head in the gas channel.

Once the capillary pressure is obtained by solving Equation 21.33 (p. 908), liquid saturation is computed from the following Leverett function (except in the membrane that is non-porous to gas species transport):

$$p_c = \sigma \left|\cos\theta_c\right| \sqrt{\frac{\varepsilon}{K}}\, J(s) \tag{21.36}$$

$$J(x) = ax - bx^2 + cx^3 \tag{21.37}$$

where σ is the surface tension (N/m), θ_c is the contact angle, and a, b, and c are the user-specified Leverett function coefficients with the following default values:

$$a = 1.417,\ b = 2.12,\ c = 1.263$$

Liquid water will reduce the effective active surface area in the catalyst layers. This is modeled by modifying the transfer currents as follows:

$$R_j = \left(1 - s\right)^{\gamma_j} R_j \tag{21.38}$$

where γ_j is a user-specified constant.

21.1.4.2.2. Liquid Water Transport Equation in Gas Channels

Liquid water leaves the gas diffusion layers and enters the gas channels. The main purpose of modeling the presence of liquid water in gas channels is to predict the pressure drop increase. In the PEMFC model, liquid water in the channels is tracked using the following correlation:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$\frac{\partial}{\partial t}\left(\rho_l s\right)+\nabla\cdot\left(\rho_l\vec{v}_l s\right)=\nabla\cdot\left(D_{liq}\nabla s\right) \tag{21.39}$$

where D_{liq} is the liquid water diffusion coefficient in the gas channel, and $\vec{v}_l$ is the liquid velocity which is assumed to be a fraction of the gas velocity $\vec{v}_g$:

$$\vec{v}_l=\chi\vec{v}_g \tag{21.40}$$

where χ is the liquid to gas velocity ratio.

At the anode and cathode flow inlets, liquid saturation $s = 0$. The liquid flux calculated from Equation 21.35 (p. 909) is used as a boundary condition at the GDL-channel interfaces for Equation 21.39 (p. 910). Since it is reasonable to assume that the flow is convection-dominated, the phase change in the gas channel is not considered here. With some meaningful level of saturation in the gas channels, momentum resistance can be constructed to model the pressure drop using the UDF function `resistance_in_channel (real sat)` in `pemfc_user.c`.

21.1.5. Heat Source

Additional volumetric sources in the thermal energy equation are present because not all chemical energy released in the electrochemical reactions can be converted to electrical work due to phase changes and irreversibilities of the processes. Volumetric heat source terms in various zones are listed in Table 21.1: Volumetric Heat Source Terms (p. 910) ([583] (p. 1086)).

Table 21.1: Volumetric Heat Source Terms

Zone	Additional Source Term
GDL+MPL	$i_s^2/\sigma_{sol}-S_{gl}\cdot L$
Anode Catalyst Layer	$R_{an}\left(\eta_{an}-\frac{T\Delta S_{an}}{2F}\right)+\frac{i_s^2}{\sigma_{sol}}+\frac{i_m^2}{\sigma_{mem}}-\left(S_{dl}+S_{gl}\right)\cdot L$
Cathode Catalyst Layer	$R_{cat}\left(-\eta_{cat}-\frac{T\Delta S_{cat}}{2F}\right)+\frac{i_s^2}{\sigma_{sol}}+\frac{i_m^2}{\sigma_{mem}}-\left(S_{dl}+S_{gl}\right)\cdot L$
Membrane (solid)	i_m^2/σ_{mem}
Current Collector (solid)	i_s^2/σ_{sol}
Gas Channels	None

In this table, i_s and i_m are the magnitude of the solid phase and membrane phase current density, respectively, and $L<0$ is the latent heat due to water condensation.

21.1.6. Properties

- Gas Phase Species Diffusivity

  Gas phase species diffusivities can be computed either by using the dilute approximation method or by using the full multicomponent method. With the dilute approximation method, we have

$$D_i=\varepsilon^{1.5}\left(1-s\right)^{r_s}D_i^0\left(\frac{p_0}{p}\right)^{\gamma_p}\left(\frac{T}{T_0}\right)^{\gamma_t} \tag{21.41}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

where ε is the porosity of the porous medium, D_i^0 is the mass diffusivity of species i at reference temperature and pressure (P_0, T_0) [660] (p. 1091). These reference values and the exponents (γ_p,γ_t) as well as the exponent of pore blockage (r_s) are defined in the PEMFC user-defined functions (UDF) as,

$$\begin{aligned} p_0 &= 101325 N/m^2 \\ T_0 &= 300 K \\ \gamma_p &= 1.0 \\ \gamma_t &= 1.5 \\ r_s &= 2.5 \end{aligned} \tag{21.42}$$

In addition to Equation 21.41 (p. 910), the Ansys Fluent PEMFC model also contains a method to compute the gas phase species diffusion (a full multicomponent diffusion method with corrections to account for the porous media tortuosity):

$$D_{eff}^{ij} = \frac{\left(1-s\right)^{r_s} \varepsilon^{1.5}}{1/D^{ij} + 1/D_{k,ij}} \tag{21.43}$$

where D_{eff}^{ij} is the effective gas species diffusivity, D^{ii} is the gas species mass diffusivity computed by the full multicomponent diffusion method (see Full Multicomponent Diffusion in the *Fluent Theory Guide* (p. 239)), and $D_{k,ij}$ is the Knudsen diffusivity computed from Equation 21.101 (p. 927). Note that the Knudsen diffusion plays an important role in determining the gas diffusivity because the average pore size in the porous media can be on the same order as the mean free path in PEMFC.

Note that $\varepsilon^{1.5}$ in Equation 21.43 (p. 911) is used to model the effect of tortuosity. While this is implemented as the default method in the PEMFC, you can overwrite it with your own correction methods by using the user-modifiable routines that are provided.

Properties such as electrolyte phase electrical conductivity, water diffusivity, and the osmotic drag coefficient are evaluated as functions of the water content, using various correlations as suggested by [624] (p. 1089). To capture the relevant physics of the problem, various properties of the membrane are incorporated into the model as default options. You can, however, directly incorporate your own formulations and data for these properties by editing the functions defined in the provided source code file called `pem_user.c` and compiling the code yourself. For more information, see User-Accessible Functions.

- Electrolyte Phase (Ionic) Conductivity

  The electrolyte (also called the membrane) phase conductivity is modeled based upon [624] (p. 1089):

  $$\sigma_{mem} = \Gamma_i \left(0.514\lambda - 0.326\right)^{\omega_i} e^{E_i\left(\frac{1}{303} - \frac{1}{T}\right)} \tag{21.44}$$

  where λ is the water content, and E_i is the activation energy for the temperature correction term. The Γ_i is calculated as ([583] (p. 1086)):

$$\Gamma_i = \begin{cases} \beta_{mem} & in\ membrane \\ \beta_{an}\frac{\varsigma_a}{\tau_a} & in\ anode\ catalyst \\ \beta_{ca}\frac{\varsigma_c}{\tau_c} & in\ cathode\ catalyst \end{cases} \tag{21.45}$$

where

ς_a = anode ionomer volume fraction

ς_c = cathode ionomer volume fraction

τ_a = anode ionomer tortuosity

τ_c = cathode ionomer tortuosity

Two model constants, β and ω are introduced in Ansys Fluent for generality. Equation 21.44 (p. 911) becomes the original correlation from [624] (p. 1089) when $\beta=\omega=1$.

- Diffusivity of Water Content

  The diffusivity coefficient in the water content (dissolved phase) transport equation is computed as follows:

$$D_w^i = \eta_\lambda \frac{\rho_i}{EW} f(\lambda) \tag{21.46}$$

  where η_λ is a user-specified coefficient for generality, and the function $f(\lambda)$ can be computed from Wu et al. [715] (p. 1094) or Wang and Wang [688] (p. 1093):

  – Wu et al:

$$f(\lambda) = 4.1e^{-10}\left(\frac{\lambda}{25}\right)^{0.15}\left[1+\tanh\left(\frac{\lambda-2.5}{1.4}\right)\right] \tag{21.47}$$

  – Wang and Wang:

$$f(\lambda) = \begin{cases} 3.1\times10^{-3}\lambda\left(e^{0.28\lambda}-1\right)e^{\left[-2346/T\right]} & for\ 0<\lambda\leq 3 \\ 4.17\times10^{-4}\lambda\left(1+161e^{-\lambda}\right)e^{\left[-2346/T\right]} & otherwise \end{cases} \tag{21.48}$$

- Osmotic Drag Coefficient

$$n_d = \eta_{osm} g(\lambda) \tag{21.49}$$

  where η_{osm} is a user-specified coefficient for generality with the default value of 1.0. The default formulation for $g(\lambda)$ is:

$$g(\lambda) = 2.5\frac{\lambda}{22}$$

  This formulation can be changed using the provided user-accessible function code file `pemfc_user.c`.

- Saturation Pressure

  The saturation pressure is calculated, in terms of atm, as,

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$\begin{aligned} log_{10}P_{sat} = & -2.1794+0.02953\left(T-273.17\right) \\ & -9.1837\times10^{-5}\left(T-273.17\right)^{2} \\ & +1.4454\times10^{-7}\left(T-273.17\right)^{3} \end{aligned} \tag{21.50}$$

Ansys Fluent allows you to provide a custom formulation by modifying the user-accessible function code file `pemfc_user.c`.

21.1.7. Transient Simulations

Dynamics response to changes in operating conditions as a function of time can be modeled using the PEM Fuel Cell module. Examples include modeling a change in the cell voltage or current density, or inlet mass flow rates at the anode and/or the cathode. The procedure for setting up and solving transient PEM Fuel Cell problems is the same as that used for a normal Ansys Fluent transient problem as discussed in the Ansys Fluent User's Guide.

Assuming that the time scales associated with the electric fields are much smaller than those associated with the flow and thermal fields, the steady-state equations are retained for the two electric potentials (Equation 21.3 (p. 901) and Equation 21.4 (p. 901)). Transient terms in all other equations such as momentum transport, energy transport, species transport, dissolved water transport (Equation 21.25 (p. 907)), liquid water transport (the capillary pressure Equation 21.33 (p. 908)), and the liquid saturation in gas channels (Equation 21.39 (p. 910)), are activated.

21.1.8. Leakage Current (Cross-Over Current)

The effect of hydrogen cross-over from anode to cathode through the membrane is modeled by a total leakage current I_l (A). In addition to the source terms expressed by Equation 21.18 (p. 905) through Equation 21.20 (p. 905), the following extra volumetric source terms are added to the corresponding equations:

$$S_{H_2} = -\frac{M_{w,H_2}}{2F}\cdot\frac{I_l}{Vol_{anode}} \tag{21.51}$$

$$S_{O_2} = -\frac{M_{w,O_2}}{4F}\cdot\frac{I_l}{Vol_{cathode}} \tag{21.52}$$

$$S_{\lambda} = \frac{M_{w,H_2O}}{2F}\cdot\frac{I_l}{Vol_{cathode}} \tag{21.53}$$

Here, $Vol_{cathode}$ and Vol_{anode} are the volumes of the catalytic layers on the anode and cathode sides, and 2 and 4 are the numbers of electrons per mole of reactants and products.

Accordingly, the volumetric leakage current ($I_l / Vol_{cathode}$) is subtracted from the cathode transfer current (R_{cat}) that is computed by Equation 21.6 (p. 903) and Equation 21.8 (p. 903).

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

21.1.9. Zones Where User-Defined Scalars Have Physically Meaningful Values

Table 21.2: Zones where UDSs are Solved in PEMFC (p. 914) lists the polymer electrolyte membrane fuel cell zones where electric potential (UDS-0), protonic potential (UDS-1), capillary pressure (UDS-2), water content (UDS-3), and liquid saturation in channels (UDS-4) have physically meaningful values.

Table 21.2: Zones where UDSs are Solved in PEMFC

Anode to Cathode	Zone Type	Electric Potential (UDS-0)	Protonic Potential (UDS-1)	Capillary Pressure (UDS-2)	Water Content (UDS-3)	Liquid Saturation in Channels (UDS-4)
Current Collector	Solid	yes				
Gas Channel	Fluid					yes
GDL	Fluid	yes		yes		
MPL	Fluid	yes		yes		
Catalyst Layer	Fluid	yes	yes	yes	yes	
Membrane	Solid		yes	yes	yes	
Catalyst Layer	Fluid	yes	yes	yes	yes	
MPL	Fluid	yes		yes		
GDL	Fluid	yes		yes		
Gas Channel	Fluid					yes
Current Collector	Solid	yes				

21.2. Fuel Cell and Electrolysis Model Theory

This chapter presents the theoretical background for the Fuel Cell and Electrolysis modeling capabilities in Ansys Fluent.

21.2.1. Introduction
21.2.2. Electrochemistry Modeling
21.2.3. Current and Mass Conservation
21.2.4. Heat Source
21.2.5. Liquid Water Formation, Transport, and its Effects (Low-Temperature PEMFC Only)
21.2.6. Properties
21.2.7. Transient Simulations
21.2.8. Leakage Current (Cross-Over Current)

21.2.1. Introduction

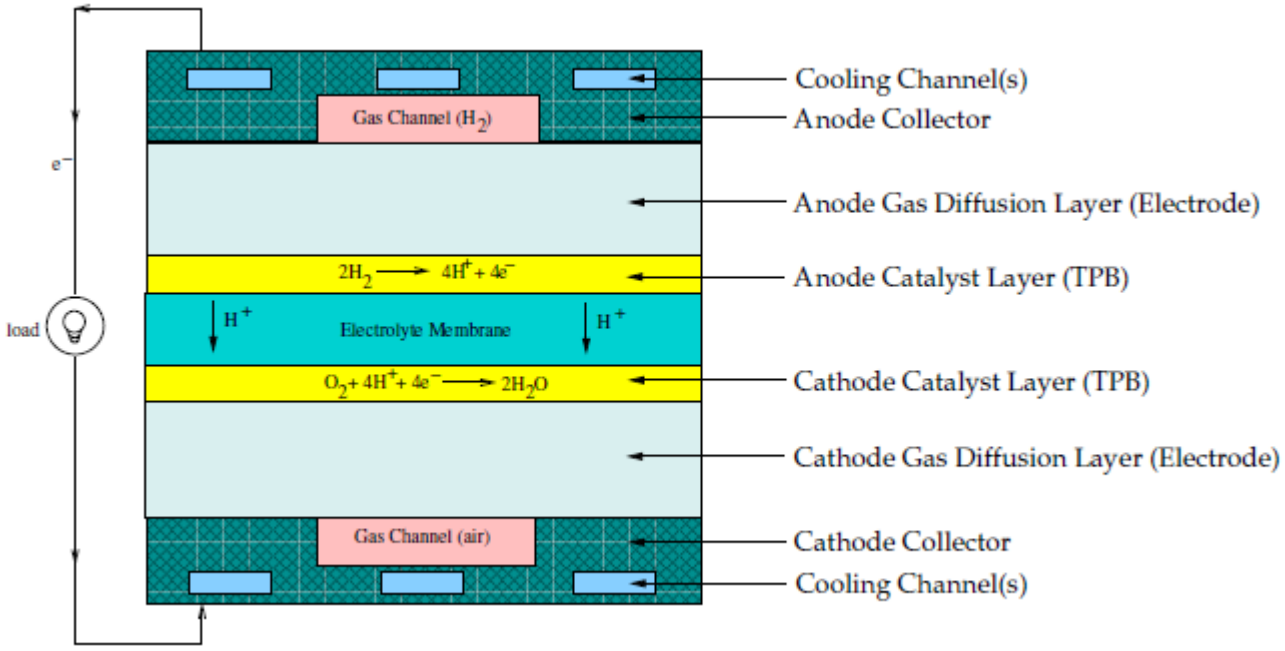
The Fuel Cell and Electrolysis module (sometimes referred to as the Resolved Electrolyte module) is provided as an add-on module with the standard Ansys Fluent licensed software.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

A fuel cell is an energy conversion device that converts the chemical energy of fuel into electrical energy. With the Fuel Cell and Electrolysis Model, both the triple-phase boundary (TPB), also known as the catalyst layer, and the ionic conducting electrolyte (also known as the membrane in PEMFC terminology) are included in the computational domain. The Fuel Cell and Electrolysis module allows you to model PEMFC, SOFC, and high-temperature electrolysis.

To determine the physical domains that are included in the Fuel Cell and Electrolysis module, a schematic of a polymer electrolyte membrane fuel cell (PEMFC) is shown in Figure 21.3: Schematic of a PEM Fuel Cell (p. 915).

Figure 21.3: Schematic of a PEM Fuel Cell



Hydrogen flows into the fuel cell on the anode side. It diffuses through the porous gas diffusion layers and comes in contact with the catalyst layer. Here it forms hydrogen ions and electrons. The hydrogen ions diffuse through the polymer electrolyte membrane at the center, the electrons flow through the gas diffusion layer to the current collectors and into the electric load attached. Electrons enter the cathode side through the current collectors and the gas diffusion layer. At the catalyst layer on the cathode side, the electrons, the hydrogen ions, and the oxygen combine to form water.

In the Fuel Cell and Electrolysis Model in Ansys Fluent, two electric potential fields are solved. One potential is solved in the electrolyte and the TPB catalyst layer. The other is solved in the TPB catalyst layer, the porous electrode, and the current collectors. The rate of electrochemical reactions are computed in the TPB layers at both the anode and the cathode. Based on the cell voltage that you prescribe, the current density value is computed. Alternatively, a cell voltage can be computed based on a prescribed average current density.

For more information, see the following sections:

21.2.1.1. Introduction to PEMFC

21.2.1.2. Introduction to SOFC

21.2.1.3. Introduction to Electrolysis

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

21.2.1.1. Introduction to PEMFC

21.2.1.1.1. Low-Temperature PEMFC

The proton exchange membrane fuel cell (PEMFC) has emerged as a favored technology for auto transportation and power generation because it is compact, clean, runs at low temperature ($<100^{\circ}$C), permits an adjustable power output, and can be started relatively rapidly. Hydrogen is supplied at the anode and air is supplied at the cathode. The following electrochemical reactions take place in the anode and cathode triple phase boundary (TPB) layers, respectively,

$$H_2 \Leftrightarrow 2H^+ + 2e^- \quad (anode\ TPB) \tag{21.54}$$

$$\frac{1}{2}O_2 + 2e^- + 2H^+ \Leftrightarrow H_2O \quad (cathode\ TPB) \tag{21.55}$$

Electrons produced in the anode travel through an external circuit to the cathode, while protons (H^+) travel through the membrane from the anode TPB to the cathode TPB, thereby forming an electrical circuit.

As more and more water is generated at the cathode, due to both osmotic drag and electrochemical reactions, water vapor pressure exceeds saturation pressure forming liquid water. The formation and transport of liquid water in the cathode is an important feature that can strongly influence cell performance of a PEMFC.

21.2.1.1.2. High-Temperature PEMFC

While the conventional PEMFC operates at low temperatures to keep the water-based membrane humidified for the purpose of allowing proton transport across the electrodes, the high-temperature PEMFC uses mineral acid-based membrane. This type of PEMFC can operate at temperatures well above 100° C. The main advantage of such a device is the elimination of humidifiers since water is present only in gas phase at the high temperature.

21.2.1.2. Introduction to SOFC

Unlike the low-temperature PEMFC, solid oxide fuel cells operate at very high temperatures ($\sim 1000^{\circ}$C). Hydrogen is supplied at the anode, either directly or through internal reforming of another hydrocarbon fuel, and air is supplied at the cathode. The following electrochemical reactions take place in the anode and cathode triple phase boundary layers, respectively,

$$H_2 + O^{2-} \Leftrightarrow H_2O + 2e^- \quad (anode\ TPB) \tag{21.56}$$

$$\frac{1}{2}O_2 + 2e^- \Leftrightarrow O^{2-} \quad (cathode\ TPB) \tag{21.57}$$

In the cathode TPB, oxygen is reduced to oxygen ions which are then conducted through the ceramic electrolyte to the anode TPB where they react with hydrogen to form water and release electrons. The electrons travel through an external circuit to a load and then back to the cathode to close the circuit.

21.2.1.3. Introduction to Electrolysis

There has been an increasingly strong need for the large-scale production of hydrogen as a secondary energy carrier. One of the cleaner and more efficient methods of producing hydrogen is to use high-temperature electrolysis to split water molecules. Therefore, electrolysis is essentially a reversed

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

fuel cell process. Power is supplied to an electrolyzer to convert water vapor into hydrogen and oxygen. Water vapor is fed through the anode electrodes to the active electrolyte region. The following electrochemical reactions take place:

$$H_2O+2e^- \Leftrightarrow H_2+O^{2-} \quad (anode\ TPB) \tag{21.58}$$

$$O^{2-} \Leftrightarrow \tfrac{1}{2}O_2+2e^- \quad (cathode\ TPB) \tag{21.59}$$

Note that, conventionally, the negative voltage is supplied to the cathode side in power consuming devices such as electrolyzers. Ansys Fluent adopts the inverse notation where the negative voltage is supplied to the anode side while cathode remains positively charged. The main reason for this discrepancy is that the same infrastructure is used for both the electrolysis and fuel cells models. Usage of the terms "anode" and "cathode" in this manual and in the user interface should be interpreted according to conventions for power-supplying devices.

In electrolysis, the activation overpotentials have the opposite sign of what is used in fuel cells. This means that the cell voltage is higher than the open circuit voltage, since power is added to overcome the activation overpotentials. The ionic conductivity in the electrolyte is typically a function of temperature, such as in the case of SOFC. And it is pointed out here that, for an electrolyzer, high thermodynamic efficiency can be achieved only at a high operating temperature ($>500^{\circ}$ C). Because of this, the flow field is in vapor phase only and is handled as such within Ansys Fluent.

21.2.2. Electrochemistry Modeling

At the center of the electrochemistry is the computation of the rates of the anodic and cathodic reactions. The electrochemistry model adopted in Ansys Fluent is one that has been used by other groups ([323] (p. 1071), [416] (p. 1077), and [660] (p. 1091)).

The driving force behind these reactions is the surface overpotential: the difference between the phase potential of the solid and the phase potential of the electrolyte/membrane. Therefore, two potential equations are solved for in the Fuel Cell and Electrolysis Model: one potential equation (Equation 21.60 (p. 917)) accounts for the electron transport e^- through the solid conductive materials and is solved in the TPB catalyst layer, the solid grids of the porous media, and the current collector; the other potential equation (Equation 21.61 (p. 917)) represents the protonic (that is, ionic) transport of H^+ or O^{-2} and is solved in the TPB catalyst layer and the membrane. The two potential equations are as follows:

$$\nabla\cdot\left(\sigma_{sol}\nabla\phi_{sol}\right)+R_{sol} = 0 \tag{21.60}$$

$$\nabla\cdot\left(\sigma_{mem}\nabla\phi_{mem}\right)+R_{mem} = 0 \tag{21.61}$$

where

σ = electrical conductivity (1/ohm-m)

ϕ = electric potential (volts)

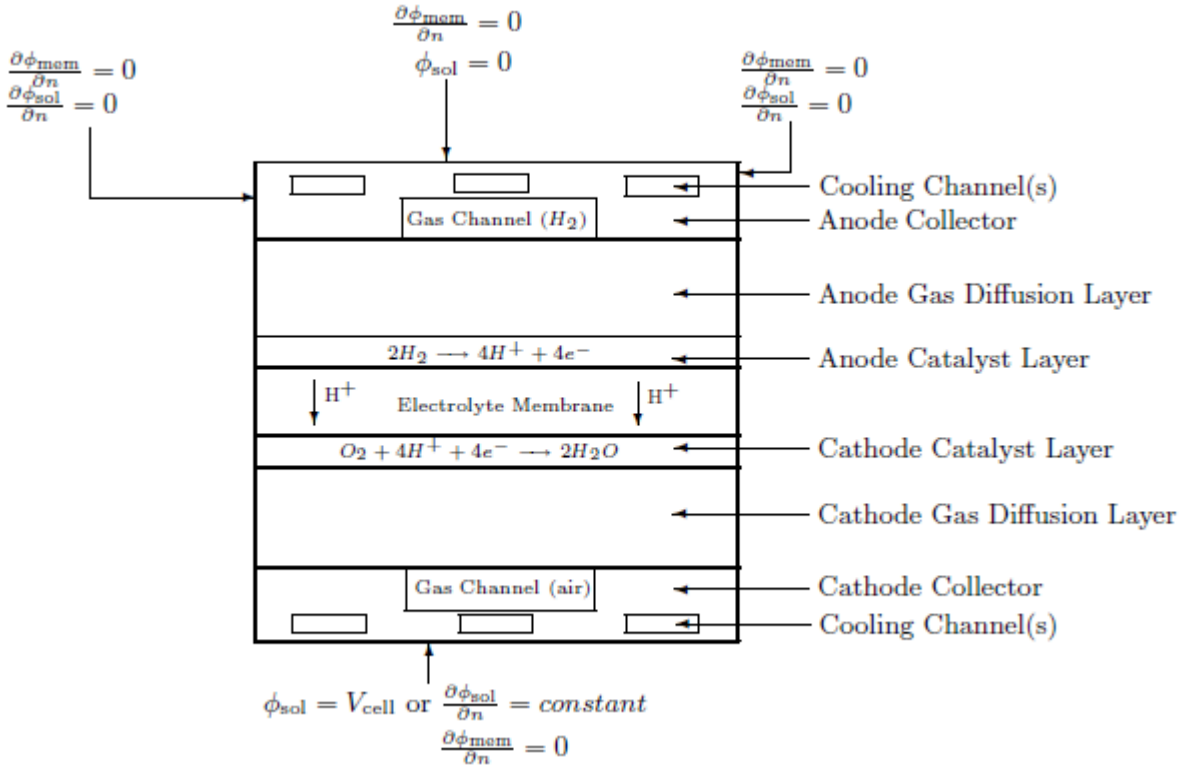
R = volumetric transfer current (A/m^3)

In the above equations, subscripts mem and sol refer to the membrane and solid phases, respectively.

The following figure illustrates the boundary conditions that are used to solve for ϕ_{sol} and ϕ_{mem}.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Figure 21.4: Boundary Conditions for the Electric Potential (Solid and Membrane) — PEM Fuel Cell



There are two types of external boundaries: those that have an electrical current passing through them, and those that do not.

As no ionic current leaves the fuel cell through any external boundary, there is a zero flux boundary condition for the membrane phase potential, ϕ_{mem}, on all outside boundaries.

For the solid phase potential, ϕ_{sol}, there are external boundaries on the anode and the cathode side that are in contact with the external electric circuit and only through these boundaries passes the electrical current generated in the fuel cell. On all other external boundaries there is a zero flux boundary condition for ϕ_{sol}.

On the external contact boundaries, we recommend to prescribe fixed values for ϕ_{sol} (potentiostatic boundary conditions). If the anode side is set to zero, the (positive) value prescribed on the cathode side is the cell voltage. Specifying a constant flux (say on the cathode side) means to specify galvanostatic boundary conditions.

The transfer currents, or the source terms in Equation 21.60 (p. 917) and Equation 21.61 (p. 917), are nonzero only inside the catalyst layers and are computed as:

- For the potential equation in the solid phase, $R_{sol}=-R_{an}\left(<0\right)$ on the anode side and $R_{sol}=+R_{cat}\left(>0\right)$ on the cathode side.
- For the potential equation in the membrane phase, $R_{mem}=+R_{an}\left(>0\right)$ on the anode side and $R_{mem}=-R_{cat}\left(<0\right)$ on the cathode side.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

The source terms in Equation 21.60 (p. 917) and Equation 21.61 (p. 917), also called the exchange current density (A/m^3), have the following general definitions:

$$R_{an} = \left(\zeta_{an} j_{an}^{ref}\right)\left(\frac{[A]}{[A]_{ref}}\right)^{\gamma_{an}}\left(e^{\alpha_{an}F\eta_{an}/RT} - e^{-\alpha_{cat}F\eta_{an}/RT}\right) \tag{21.62}$$

$$R_{cat} = \left(\zeta_{cat} j_{cat}^{ref}\right)\left(\frac{[C]}{[C]_{ref}}\right)^{\gamma_{cat}}\left(-e^{+\alpha_{an}F\eta_{cat}/RT} + e^{-\alpha_{cat}F\eta_{cat}/RT}\right) \tag{21.63}$$

where

j^{ref} = reference exchange current density per active surface area (A/m^2)

ζ = specific active surface area (1/m)

$[\,]$, $[\,]_{ref}$ = local species concentration, reference value ($\mathrm{kmol/m^3}$)

γ = concentration dependence

α = transfer coefficient (dimensionless)

F = Faraday constant (9.65×10^7 C/kmol)

The above equation is the general formulation of the Butler-Volmer function. Note that the effects of the number of electrons in electrochemistry reactions are accounted for in the transfer coefficients. A simplification to this is the Tafel formulation that reads,

$$R_{an} = \left(\zeta_{an} j_{an}^{ref}\right)\left(\frac{[A]}{[A]_{ref}}\right)^{\gamma_{an}}\left(e^{\alpha_{an}F\eta_{an}/RT}\right) \tag{21.64}$$

$$R_{cat} = \left(\zeta_{cat} j_{cat}^{ref}\right)\left(\frac{[C]}{[C]_{ref}}\right)^{\gamma_{cat}}\left(e^{-\alpha_{cat}F\eta_{cat}/RT}\right) \tag{21.65}$$

By default, the Butler-Volmer function is used in the Ansys Fluent Fuel Cell and Electrolysis Model to compute the transfer currents inside the catalyst layers.

In Equation 21.62 (p. 919) through Equation 21.65 (p. 919), $[A]$ and $[C]$ represent the molar concentration of the species upon which the anode and cathode reaction rates depend, respectively. For PEMFC and SOFC, A represents H_2 and C represents O_2. For Electrolysis, A represents H_2O and C is 1.0 (which indicates that the cathode reaction does not depend on any species concentration).

The driving force for the kinetics is the local surface overpotential, η, also known as the *activation loss*. It is generally the difference between the solid and membrane potentials, ϕ_{sol} and ϕ_{mem}.

The gain in electrical potential from crossing from the anode to the cathode side can then be taken into account by subtracting the open-circuit voltage V_{oc} on the cathode side.

$$\eta_{an} = \phi_{sol} - \phi_{mem} \tag{21.66}$$

$$\eta_{cat} = \phi_{sol} - \phi_{mem} - V_{oc} \tag{21.67}$$

From Equation 21.60 (p. 917) through Equation 21.67 (p. 919), the two potential fields can be obtained.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

21.2.3. Current and Mass Conservation

Species volumetric source terms (kg/m^3-s) in the triple-phase boundaries due to electrochemical reactions for the PEMFC, SOFC, and Electrolysis, respectively, are:

For PEMFC:

$$S_{H_2} = -\frac{M_{w,H_2}}{2F} R_{an} < 0 \tag{21.68}$$

$$S_{O_2} = -\frac{M_{w,O_2}}{4F} R_{cat} < 0 \tag{21.69}$$

$$S_{H_2O} = \frac{M_{w,H_2O}}{2F} R_{cat} > 0 \tag{21.70}$$

For SOFC:

$$S_{H_2} = -\frac{M_{w,H_2}}{2F} R_{an} < 0 \tag{21.71}$$

$$S_{O_2} = -\frac{M_{w,O_2}}{4F} R_{cat} < 0 \tag{21.72}$$

$$S_{H_2O} = \frac{M_{w,H_2O}}{2F} R_{an} > 0 \tag{21.73}$$

For Electrolysis:

$$S_{H_2} = \frac{M_{w,H_2}}{2F} R_{an} > 0 \tag{21.74}$$

$$S_{O_2} = \frac{M_{w,O_2}}{4F} R_{cat} > 0 \tag{21.75}$$

$$S_{H_2O} = -\frac{M_{w,H_2O}}{2F} R_{an} < 0 \tag{21.76}$$

In the above equations, M_{w,H_2O}, M_{w,O_2}, and M_{w,H_2} are the molecular mass of water, oxygen and hydrogen, respectively, F is the Faraday constant, and 2 and 4 are the numbers of electrons per mole of reactants and products.

Since the total electrical current produced in the cathode and the anode TPBs, respectively, is the same, we have the following equation for current conservation:

$$\int_{anode} R_{an} dV = \int_{cathode} R_{cat} dV \tag{21.77}$$

21.2.4. Heat Source

Additional volumetric sources to the thermal energy equation are present because not all chemical energy released in the electrochemical reactions can be converted to electrical work due to irreversibilities of the processes. The total source that goes to the thermal energy equation (that is, enthalpy) is:

$$S_h = h_{react} - R_{an,cat}\eta_{an,cat} + I^2 R_{ohm} + h_L \tag{21.78}$$

where h_{react} is the net enthalpy change due to the electrochemical reactions, $R_{an,cat}\eta_{an,cat}$ is the product of the transfer current and the overpotential in the anode or the cathode TPB, R_{ohm} is the ohmic resistivity of the conducting media, and h_L is the enthalpy change due to condensation/vaporization of water.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

21.2.5. Liquid Water Formation, Transport, and its Effects (Low-Temperature PEMFC Only)

Since low-temperature PEM fuel cells operate under relatively low temperature (<100°C), the water vapor may condense to liquid water, especially at high current densities. While the existence of the liquid water keeps the membrane hydrated, it also blocks the gas diffusion passage, reduces the diffusion rate and the effective reacting surface area and hence the cell performance. To model the formation and transport of liquid water, Ansys Fluent uses a saturation model based on [473] (p. 1080), [466] (p. 1080). In this approach, the liquid water formation and transport is governed by the following conservation equation for the volume fraction of liquid water, s, or the water saturation,

$$\frac{\partial(\varepsilon\rho_{\mathrm{l}}s)}{\partial t}+\nabla\cdot\left(\rho_{\mathrm{l}}\vec{V}_{\mathrm{l}}s\right)=r_{\mathrm{w}} \tag{21.79}$$

where ε is the porosity, the subscript l stands for liquid water, and r_w is the condensation rate that is modeled as,

$$r_w=\begin{cases}(1-\mathrm{s})\,c_r\frac{P_{wv}-P_{sat}}{RT}M_{h2o} & if\ (P_{wv}>P_{sat})\\ s\,c_r\frac{P_{wv}-P_{sat}}{RT}M_{h2o} & if\ (P_{wv}<P_{sat})\end{cases} \tag{21.80}$$

where $-r_w$ is added to the water vapor equation, as well as the pressure correction (mass source). This term is applied only inside the catalyst and gas diffusion layers. The model accounts for the phase change on the cathode side only. An excessive water vapor generation due to the electrochemical reaction can lead to condensation on the cathode side at higher loads. Condensation due to normal causes, such as lower temperatures and higher pressures on the anode side, is rare and not accounted for. When condensation considerations are important for your simulation, use a more general PEMFC model described in PEMFC Model Theory (p. 899). The model accounts for condensation across all fuel cell layers including both gas channels. More details about the model implementation can be found in Water Transport and Mass Transfer in PEMFC (p. 906).

The condensation rate constant is hardwired to $c_r=100s^{-1}$. It is assumed that the liquid velocity, V_l, is equivalent to the gas velocity inside the gas channel (that is, a fine mist). Inside the highly-resistant porous zones, the use of the capillary diffusion term allows us to replace the convective term in Equation 21.79 (p. 921):

$$\frac{\partial(\varepsilon\rho_l s)}{\partial t}+\nabla\cdot\left[\rho_l\frac{Ks^{\gamma}}{\mu_l}\frac{dp_c}{ds}\nabla s\right]=r_w \tag{21.81}$$

Depending on the wetting phase, the capillary pressure is computed as a function of s (the Leverett function) [356] (p. 1073):

$$p_c=\begin{cases}\frac{\sigma\cos\theta_c}{\left(\frac{K}{\varepsilon}\right)^{0.5}}\left(1.417\left(1-s\right)-2.12\left(1-s\right)^2+1.263\left(1-s\right)^3\right) & \theta_c<90^\circ\\ \frac{\sigma\cos\theta_c}{\left(\frac{K}{\varepsilon}\right)^{0.5}}\left(1.417s-2.12s^2+1.263s^3\right) & \theta_c>90^\circ\end{cases} \tag{21.82}$$

where σ is the surface tension (N/m), K the absolute permeability, and θ_c is the contact angle. Values of θ_c from 0 to 90 degrees represent a hydrophilic media, while values from 90 to 180 degrees represent a hydrophobic media.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Equation 21.79 (p. 921) models various physical processes such as condensation, vaporization, capillary diffusion, and surface tension.

The clogging of the porous media and the flooding of the reaction surface are modeled by multiplying the porosity and the active surface area by $(1-s)$, respectively.

21.2.6. Properties

- Gas Phase Species Diffusivity

  Gas phase species diffusivities can be computed either by using the dilute approximation method or by using the full multicomponent method. With the dilute approximation method, we have

$$D_i = \varepsilon^{1.5}(1-s)^{r_s} D_i^0 \left(\frac{p_0}{p}\right)^{\gamma_p} \left(\frac{T}{T_0}\right)^{\gamma_t} \tag{21.83}$$

  where D_i^0 is the mass diffusivity of species i at reference temperature and pressure (P_0, T_0) [660] (p. 1091). These reference values and the exponents (γ_p,γ_t) as well as the exponent of pore blockage (r_s) are defined in the Fuel Cell and Electrolysis user defined functions (UDF) as,

$$\begin{aligned} p_0 &= 101325 N/m^2 \\ T_0 &= 300 K \\ \gamma_p &= 1.0 \\ \gamma_t &= 1.5 \\ r_s &= 2.5 \end{aligned} \tag{21.84}$$

  In addition to Equation 21.83 (p. 922), the Ansys Fluent Fuel Cell and Electrolysis Model also contains a method to compute the gas phase species diffusion (a full multicomponent diffusion method with corrections to account for the porous media tortuosity):

$$D_{eff}^{ij} = \frac{\varepsilon^{1.5}(1-s)^{r_s}}{\left(1/D^{ij} + 1/D_{k,ij}\right)} \tag{21.85}$$

  where D_{eff}^{ij} is the effective gas species diffusivity, ε is the porosity of the porous medium, D^{ij} is the gas species mass diffusivity computed by the full multicomponent diffusion method (see Full Multicomponent Diffusion in the *Fluent Theory Guide* (p. 239)), and $D_{k,ij}$ is the Knudsen diffusivity computed from Equation 21.101 (p. 927). Note that the Knudsen diffusion plays an important role in determining the gas diffusivity because the average pore size in the porous media can be on the same order as the mean free path in the Fuel Cell and Electrolysis Model. $\varepsilon^{1.5}$ in Equation 21.85 (p. 922) is used to model the effect of tortuosity. While this is implemented as the default method in the Fuel Cell and Electrolysis Model, you can overwrite it with your own correction methods by using the user-modifiable routines that are provided.

  Properties such as electrolyte phase electrical conductivity, water diffusivity, and the osmotic drag coefficient are evaluated as functions of the water content, using various correlations as suggested by [624] (p. 1089). To capture the relevant physics of the problem, various properties of the membrane are incorporated into the model as default options. You can, however, directly incorporate your own formulations and data for these properties by editing the functions defined in the provided source code file called `pem_user.c` and compiling the code yourself. For more information, see User-Accessible Functions.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

- Electrolyte Phase (Ionic) Conductivity

  For SOFC and high-temperature Electrolysis, the ionic conductivity in the electrolyte is modeled as a function of temperature, and, by default, is defined as:

$$\sigma_{ionic}=\frac{100}{0.3685+0.002838e^{(10300/T)}} \tag{21.86}$$

  This is valid for temperatures ranging from 1073 K to 1373 K. You can implement your own models in the user-customizable UDF function `Electrolyte_Conductivity` in `pem_user.c`.

  For high-temperature PEMFC, the electrolyte (also called the membrane) phase conductivity is a user-specified constant.

  For low-temperature PEMFC, the electrolyte (also called the membrane) phase conductivity is modeled as:

$$\sigma_{mem}=\beta\left(0.514\lambda-0.326\right)^{\omega}e^{1268\left(\frac{1}{303}-\frac{1}{T}\right)} \tag{21.87}$$

  where λ is the water content. Two model constants, β and ω are introduced in Ansys Fluent for generality. Equation 21.87 (p. 923) becomes the original correlation from [624] (p. 1089) when $\beta=\omega=1$.

- Osmotic Drag Coefficient (low-temperature PEMFC)

$$n_d=2.5\frac{\lambda}{22} \tag{21.88}$$

- Back Diffusion Flux (low-temperature PEMFC)

$$J_w^{diff}=-\frac{\rho_m}{M_m}M_{h_2O}D_l\nabla\lambda \tag{21.89}$$

  where ρ_m and M_m are the density and the equivalent weight of the dry membrane, respectively.

- Membrane Water Diffusivity (low-temperature PEMFC)

$$D_l=f\left(\lambda\right)e^{2416\left(\frac{1}{303}-\frac{1}{T}\right)} \tag{21.90}$$

- Water Content (low-temperature PEMFC)

  The water content, λ, that appears in the preceding property computations are obtained using Springer et al's correlation [624] (p. 1089),

$$\begin{aligned}\lambda &= 0.043+17.18a-39.85a^2+36a^3\left(a<1\right)\\ \lambda &= 14+1.4\left(a-1\right)\left(a>1\right)\end{aligned} \tag{21.91}$$

  here a is the water activity that is defined as,

$$a=\frac{P_{wv}}{P_{sat}}+2s \tag{21.92}$$

  where P_{wv} and P_{sat} are the water vapor pressure and the saturation pressure, respectively.

- Water Vapor Pressure

  The water vapor pressure is computed based upon the vapor molar fraction and the local pressure,

$$P_{wv}=x_{H_2O}P \tag{21.93}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

- Saturation Pressure

  The saturation pressure is calculated, in terms of atm, as,

$$\begin{aligned} log_{10}P_{sat} = & -2.1794+0.02953\left(T-273.17\right) \\ & -9.1837\times10^{-5}\left(T-273.17\right)^{2} \\ & +1.4454\times10^{-7}\left(T-273.17\right)^{3} \end{aligned} \tag{21.94}$$

  It is noted here that in [624] (p. 1089), water activity is defined on the basis of total water or super-saturated water vapor. With phase change being invoked in the present two-phase model, $2s$ is added to the original formulation as suggested by [156] (p. 1062).

21.2.7. Transient Simulations

Dynamics response to changes in operating conditions as a function of time can be modeled using the Fuel Cell and Electrolysis module. For example, a change in the cell voltage or current density, or inlet mass flow rates at the anode and/or the cathode. The procedure for setting up and solving transient Fuel Cell and Electrolysis problems are the same as that used for a normal Ansys Fluent transient problem as discussed in the Ansys Fluent User's Guide.

Assuming that the time scales associated with the electric fields are much smaller than those associated with the flow and thermal fields, the steady-state equations are retained for the two electric potentials, (that is, Equation 21.60 (p. 917) and Equation 21.61 (p. 917)). Transient terms in all other equations such as momentum transport, energy transport, species transport, liquid water transport, and membrane water content equations are activated.

21.2.8. Leakage Current (Cross-Over Current)

The leakage current, I_{leak} (A), models the overall effect of species cross-over from one electrode to another across the electrolyte. In addition to the source terms expressed by Equation 21.68 (p. 920) through Equation 21.76 (p. 920), the following extra volumetric source terms are added to the corresponding equations:

For PEMFC and SOFC:

$$S_{H_2} = -\frac{M_{w,H_2}}{2F}\cdot\frac{I_{leak}}{Vol_{anode}} \tag{21.95}$$

$$S_{O_2} = -\frac{M_{w,O_2}}{4F}\cdot\frac{I_{leak}}{Vol_{cathode}} \tag{21.96}$$

$$S_{H_2O} = -\frac{M_{w,H_2O}}{2F}\cdot\frac{I_{leak}}{Vol_{cathode}} \tag{21.97}$$

For Electrolysis:

$$S_{H_2} = \frac{M_{w,H_2}}{2F}\cdot\frac{I_{leak}}{Vol_{anode}} \tag{21.98}$$

$$S_{O_2} = \frac{M_{w,O_2}}{4F}\cdot\frac{I_{leak}}{Vol_{cathode}} \tag{21.99}$$

$$S_{H_2O} = -\frac{M_{w,H_2O}}{2F}\cdot\frac{I_{leak}}{Vol_{anode}} \tag{21.100}$$

Here, $Vol_{cathode}$ and Vol_{anode} are the he volumes of the catalytic layers on the anode and cathode sides, and 2 and 4 are the numbers of electrons per mole of reactants and products.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

21.3. SOFC Fuel Cell With Unresolved Electrolyte Model Theory

This chapter presents an overview of theory and equations for solid oxide fuel cell (SOFC, with unresolved electrolyte) modeling capabilities in Ansys Fluent.

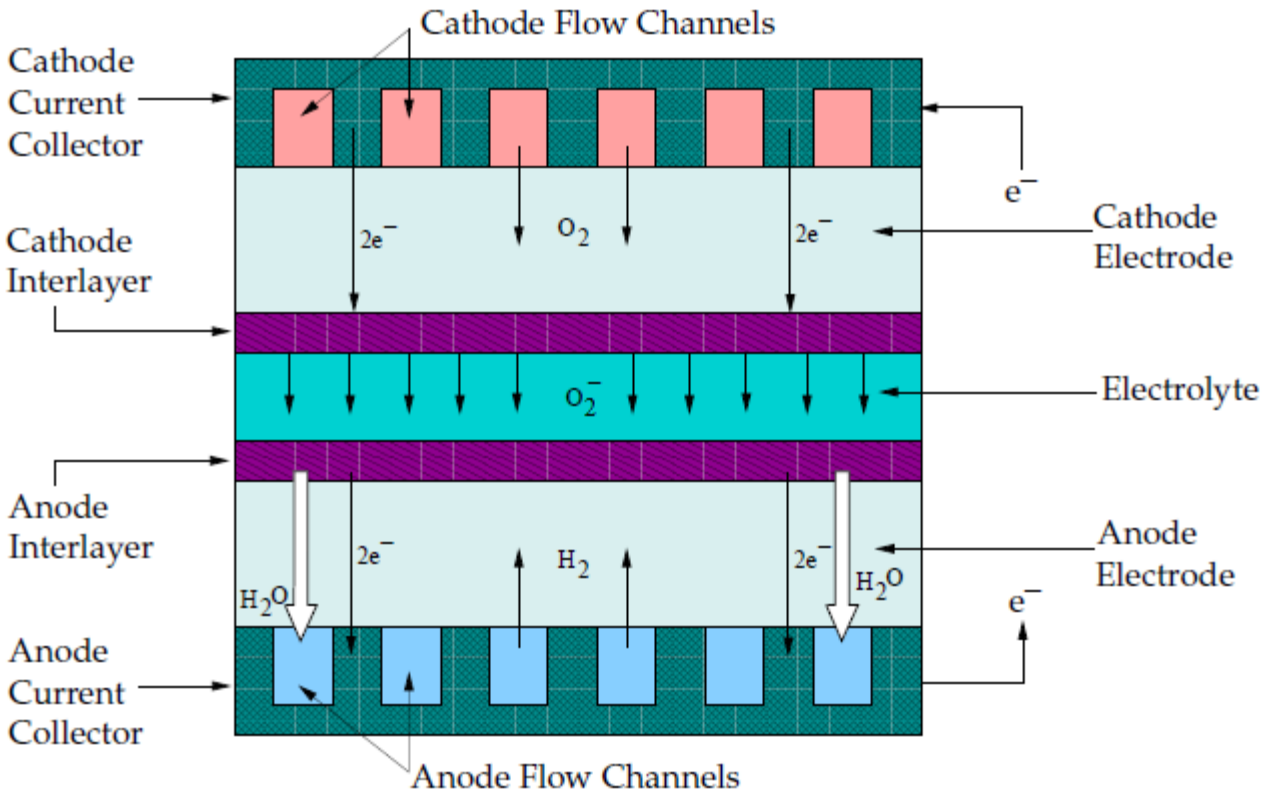
21.3.1. Introduction
21.3.2. The SOFC With Unresolved Electrolyte Modeling Strategy
21.3.3. Modeling Fluid Flow, Heat Transfer, and Mass Transfer
21.3.4. Modeling Current Transport and the Potential Field
21.3.5. Modeling Reactions

21.3.1. Introduction

The Solid Oxide Fuel Cell (SOFC) With Unresolved Electrolyte Model is provided as an add-on module with the standard Ansys Fluent licensed software.

A fuel cell is an energy conversion device that converts the chemical energy of fuel into the electrical energy. A schematic of a solid oxide fuel cell (SOFC) is shown in Figure 21.5: Schematic of a Solid Oxide Fuel Cell (p. 925).

Figure 21.5: Schematic of a Solid Oxide Fuel Cell



As noted in [2] (p. 1053), a solid oxide fuel cell is typically composed of an anode, cathode, and an electrolyte. Multiple fuel cells can be connected together, or stacked, using electrical interconnects. The electrolyte material must be solid, that is, non-porous, and exhibit a high ionic conductivity.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Note that the reason this modeling approach is referred to as the “SOFC Model with Unresolved Electrolyte" model is that the anode and the cathode “interlayers" and “electrolyte" (as shown in Figure 21.5: Schematic of a Solid Oxide Fuel Cell (p. 925)) are not actually included in the computational domain. They are modeled as a pair of wall and wall-shadow faces, named “electrolyte interfaces," with the species and energy sources and sinks due to the electrochemical reactions added to the adjacent computational cells.

All components of the fuel cell must have similar thermal expansion in order to minimize thermal stresses, which may cause cracking and de-lamination during thermal cycling. In addition, the components must be chemically stable in order to limit chemical interactions with other cell components.

A solid oxide fuel cell works by having electrically conducting porous ceramic electrodes attached on each side of an ionically conducting ceramic material. At the cathode/electrolyte/gas interface, also known as the triple phase boundary, oxygen is reduced to oxygen ions. The oxygen ions are conducted through the oxygen vacancies in the electrolyte to the anode side. At the anode/electrolyte/gas interface, oxygen ions combine to react with hydrogen at the anode electrode to form water and release electrons. The electrons travel through an external circuit to a load and back to the cathode electrode to close the circuit.

The Ansys Fluent SOFC With Unresolved Electrolyte Model provides the following features:

- Local electrochemical reactions coupling the electric field and the mass, species, and energy transport.
- Electric field solution in all porous and solid cell components, including ohmic heating in the bulk material.
- Ability to handle H_2 and combined CO/H_2 electrochemistry.
- Inclusion of tortuosity for porous regions
- Treatment of an arbitrary number of electrochemical cells arranged as a stack.
- Significant geometric flexibility for treating planar, tubular, and other nonstandard SOFC configuration.
- Use of non-conformal interface meshing (as a long as these interfaces are not the electrolyte interfaces).

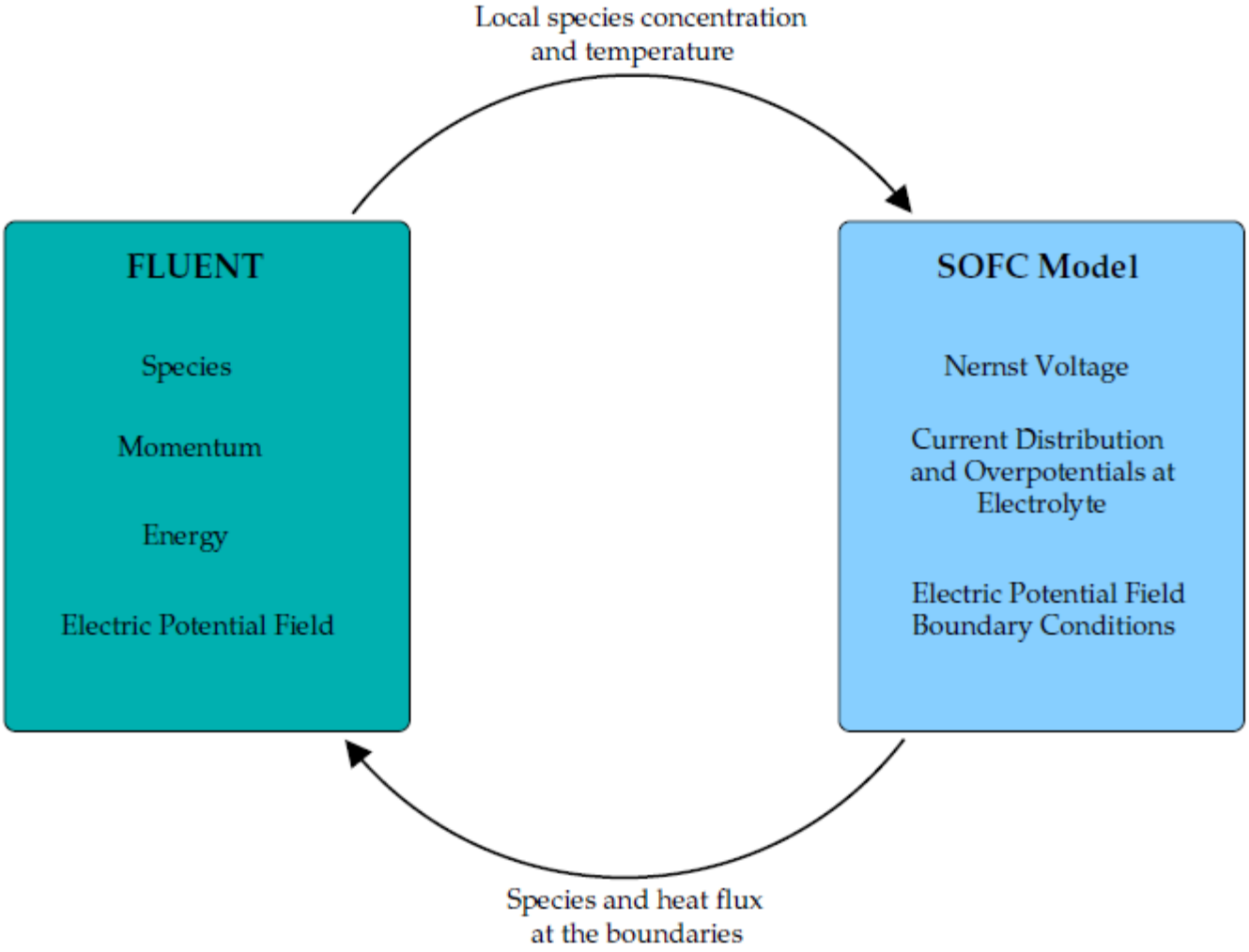
21.3.2. The SOFC With Unresolved Electrolyte Modeling Strategy

To model solid oxide fuel cells (SOFC) with unresolved electrolyte, you need to perform the following:

- Capture the fluid flow, heat transfer, and the mass transfer in the flow channels and in the porous anode and cathode electrodes.
- Model the transport of the current and the potential field in the porous electrodes and in the solid conducting regions.
- Model the electrochemical reactions that take place at the electrolyte/electrode/gaseous species interface.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Figure 21.6: How the SOFC With Unresolved Electrolyte Model Works in Ansys Fluent



21.3.3. Modeling Fluid Flow, Heat Transfer, and Mass Transfer

All aspects of fluid flow, heat transfer, and mass transfer in the flow channels and porous electrodes are handled by Ansys Fluent.

The default multicomponent diffusion model in Ansys Fluent is used to calculate the mass diffusion coefficient of species i in the mixture.

As noted in [17] (p. 1054), the average pore size of gas diffusion layer in fuel cell devices is on the same order as the length of the mean free path, making the Knudsen diffusion play a big role in computing the effective gas diffusivity.

The multi-component Knudsen diffusion coefficient $D_{k,ij}$ is computed as:

$$D_{k,ij}=\frac{2}{3}r_e\sqrt{\frac{8RT}{\pi(M_i+M_j)/2)}} \tag{21.101}$$

where

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

r_e = effective radius of the pore

R = universal gas constant

T = temperature

M_i and M_j = molecular weights of species i and j, respectively

Finally, the effective gas diffusivity coefficient is computed as:

$$D_{ij,\,eff} = \frac{\varepsilon}{\tau}\frac{1}{1/D_{ij}+1/D_{k,ij}} \tag{21.102}$$

where

τ = tortuosity (the average path length over the actual length)

ε = porosity

D_{ij} = binary diffusivity

21.3.4. Modeling Current Transport and the Potential Field

Solving for three-dimensional electrical conduction is directly analogous to the calculation of heat transfer. The potential field throughout the conductive regions is calculated based on the conservation of charge.

$$\nabla \cdot i = 0 \tag{21.103}$$

where

$$i = -\sigma \nabla \phi \tag{21.104}$$

and σ is the electrical conductivity and ϕ is the electrical potential. Therefore, the governing equation for the electric field is the Laplace equation:

$$\nabla \cdot \left(\sigma \nabla \phi\right) = 0 \tag{21.105}$$

The electric field potential calculation combines the following attributes:

- Ohmic losses in all the conducting materials, including the electrolyte, electrodes, and current collectors.
- Contact resistance at the appropriate interfaces.
- Ohmic heating through conduction materials as the result of ohmic losses, and the current density throughout the domain.

For more information, see the following sections:

21.3.4.1. Cell Potential

21.3.4.2. Activation Overpotential

21.3.4.3. Treatment of the Energy Equation at the Electrolyte Interface

21.3.4.4. Treatment of the Energy Equation in the Conducting Regions

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

21.3.4.1. Cell Potential

The electrode reactions are assumed to take place in a single step. The charge transfer reaction acts as the rate limiting step for the electrode reactions.

Oxygen is electrochemically reduced at the triple phase boundary at the cathode electrode:

$$\frac{1}{2}O_{2,cathode}+2e^{-}\Leftrightarrow O^{2-} \tag{21.106}$$

Oxygen is electrochemically re-oxidized at the triple phase boundary at the anode electrode:

$$O^{2-}\Leftrightarrow\frac{1}{2}O_{2,anode}+2e^{-} \tag{21.107}$$

In the absence of an electrical load, the oxygen activity on both sides of the electrolyte is fixed and given by their respective chemical potentials. Under equilibrium, the electromotive force, or reversible cell voltage, is given by the Nernst equation:

$$\phi_{ideal}=\frac{RT}{4F}\ln\frac{p_{O_{2,cathode}}}{p_{O_{2,anode}}} \tag{21.108}$$

If hydrogen is present at the anode electrode, then the cell reaction becomes:

$$H_2+O^{2-}\Leftrightarrow H_2O+2e^{-} \tag{21.109}$$

At equilibrium, the cell voltage is given by the Nernst equation:

$$\phi_{ideal}=\phi^{o}+\frac{RT}{2F}\ln\frac{p_{H_2}p_{O_2}^{\frac{1}{2}}}{p_{H_2O}} \tag{21.110}$$

where ϕ^{o} is the open-circuit cell voltage.

The cell potential measured at equilibrium (that is, no load), is called the open circuit voltage. The open circuit voltage should be equivalent to the Nernst potential at no load, unless there is leakage across the electrolyte. When the external circuit is closed, then cell voltage drops due to polarization losses at the electrodes.

The electric field and the electrochemistry interact solely at the electrolyte interface. Ansys Fluent treats the electrolyte interface as an impermeable wall. The potential field must have a “jump" condition applied to the two sides of this wall to account for the effect of the electrochemistry. To closely couple the electrochemical behavior to the potential field calculation, you need to include all of the electrochemical effects into this jump condition. It encapsulates the voltage jump due to Nernst, the voltage reduction due to activation, the Ohmic losses due to the resistivity of the electrolyte, and are linearized for voltage reduction due to activation. This interface condition relates the potential on the anode side and the cathode side of the electrolyte and has the following form:

$$\phi_{cell}=\phi_{jump}-\eta_s \tag{21.111}$$

where

$$\phi_{jump}=\phi_{ideal}-\eta_{ele}-\eta_{act,a}-\eta_{act,c} \tag{21.112}$$

where η_{ele} represents the ohmic overpotential of the electrolyte, and $\eta_{act,a}\eta_{act,c}$ represent the activation overpotential of the anode and the cathode. η_s represent ohmic losses in the solid conducting regions. ϕ_{ideal} represents the Nernst potential.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

21.3.4.2. Activation Overpotential

The general electrochemical reaction is, according to [470] (p. 1080),

$$\sum_{j}^{N} a_j A_j \Leftrightarrow ne^{-} \tag{21.113}$$

where a_j is the stoichiometric coefficient of species j, A_j is the chemical species, and n is the number of electrons.

Using the Bulter Volmer equation, the reaction rate (in units of kmol/(m^2 S)) can be computed as:

$$r = \frac{i}{nF} = \frac{i_{0\,eff}}{nF}\left(e^{\alpha_a n\eta_{act}F/RT} - e^{-\alpha_c n\eta_{act}F/RT}\right) \tag{21.114}$$

where i is the electric current (in units of A/m^2) generated by the electrochemical reaction, α_a is the anodic transfer coefficient, α_c is the cathodic transfer coefficient, η_{act} is the activation potential at anode and cathode, F is the Faraday constant, R is the universal gas constant, T is the temperature, and

$$i_{0\,eff} = i_{0,ref}\left(\frac{\chi_j}{\chi_{j,ref}}\right)^{\gamma_j} \tag{21.115}$$

with $i_{0,ref}$ being the exchange current density at the reference condition, $\left(\chi_j\right)$ is the mole fraction and γ_j is the concentration exponent for species j. More specifically, at the anode side, you have:

$$i_{0\,eff}^{anode} = i_{0,ref}^{anode}\left(\frac{\chi_{H_2}}{\chi_{H_2,ref}}\right)^{\gamma_{H_2}}\left(\frac{\chi_{H_2O}}{\chi_{H_2O,ref}}\right)^{\gamma_{H_2O}} \tag{21.116}$$

Likewise, at the cathode side, you have:

$$i_{0\,eff}^{cathode} = i_{0,ref}^{cathode}\left(\frac{\chi_{O_2}}{\chi_{O_2,ref}}\right)^{\gamma_{O_2}} \tag{21.117}$$

The exchange current density at both anode and cathode can consider the temperature dependency effect as:

$$i_{0,ref} = Ae^{-\frac{1}{BT}} \tag{21.118}$$

where A and B are rate coefficients.

Given values for α_a and α_c. the full version of the Butler-Volmer equation can be solved using the Newton method, therefore finding the activation overpotential at the anode ($\eta_{act,a}$) and the cathode ($\eta_{act,c}$).

21.3.4.3. Treatment of the Energy Equation at the Electrolyte Interface

For an incompressible flow, the energy equation that Ansys Fluent solves for within each computational cell is given by the following:

$$\frac{\partial}{\partial t}\left(\rho E\right) + \nabla\cdot\left(\vec{v}\left(\rho E + p\right)\right) = \nabla\cdot\left(k_{eff}\nabla T - \sum_j h_j\vec{J}_j + \left(\overline{\overline{\tau}}_{eff}\cdot\vec{v}\right)\right) + S_h \tag{21.119}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

where S_h is the volumetric source or sink of energy and where

$$E = h - \frac{p}{\rho} + \frac{v^2}{2} \tag{21.120}$$

and

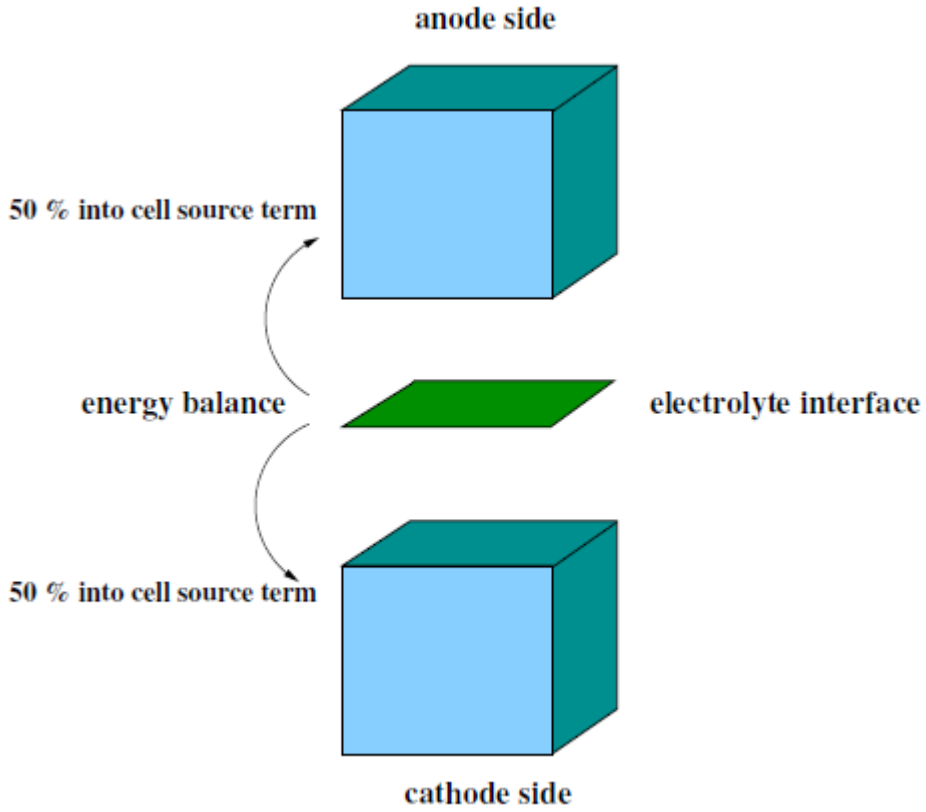
$$h = \sum_j Y_j h_j \tag{21.121}$$

In all electrically conducting zones (for example, electrodes, current collectors, interconnects), ohmic heating, $i^2 * R_{ohmic}$, is added to the energy equation as a source term. In other words,

$$S_h = i^2 * R_{ohmic} \tag{21.122}$$

In addition, the energy equation needs treatment at the electrode-electrolyte interface to account for the heat generated or lost as the result of electrochemistry and the overpotentials (that is, activation overpotential and ohmic loss through the electrolyte).

Figure 21.7: Energy Balance at the Electrolyte Interface



The energy balance on the electrolyte interface is computed by subtracting the electric work from the total heat generation rate due to electrochemical reactions. The electric work is calculated as the product of voltage jump and the current density. What remains is the waste heat due to irreversibilities. For hydrogen reaction, the balance would be

$$\dot{Q}'' = h''_{H_2} + h''_{O_2} - h''_{H_2O} - i\Delta V \tag{21.123}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

where $\dot{Q}''$ is the heat generation (W) and h'' is the product of species mass generation/consumption rate and its formation enthalpy. For example:

$$h''_{H_2} = \dot{m}_{H_2} h''_{0,H_2} \tag{21.124}$$

The source term is then added in the cell energy equation by taking $S_h = \frac{Q}{Volume}$.

One half of this value is applied as a source term to the energy equation of the anode computational cell adjacent to the electrolyte and the other half is applied as a source term to the energy equation for the cathode cell adjacent to the electrolyte. The equal distribution of the heat generation/destruction is purely arbitrary. Note that by using the work term, the effect from all overpotentials are taken into account.

21.3.4.4. Treatment of the Energy Equation in the Conducting Regions

Ohmic polarization involves ionic losses through the electrolyte, electrical resistance in the conducting porous electrodes and solid collectors. The ohmic polarization also includes the electrical resistance at the interface of the current collectors and the electrodes or the electrodes and the membrane (that is, the contact resistance).

$$\eta_{ohmic} = i \cdot R \tag{21.125}$$

21.3.5. Modeling Reactions

The Ansys Fluent SOFC With Unresolved Electrolyte Model can model both electrochemical reactions, as well as CO electrochemistry. Reforming reactions that generate hydrogen can be modeled by the standard volumetric reaction mechanism in Ansys Fluent.

For more information, see the following sections:

21.3.5.1. Modeling Electrochemical Reactions

21.3.5.2. Modeling CO Electrochemistry

21.3.5.3. Modeling Electrolysis

21.3.5.1. Modeling Electrochemical Reactions

The rate of species production and destruction is:

$$S = -\frac{ai}{nF}\left(g-mole/\ m^2/\ sec\right) \tag{21.126}$$

where S is the source or sink of the species (molar flux), a is the stoichiometric coefficient, i is the current density ($A/\ m^2$), n is the number of electrons per mole of fuel, and F is the Faraday constant.

Using the local current information, the Ansys Fluent SOFC With Unresolved Electrolyte Model applies species fluxes to the electrode boundaries. By convention [470] (p. 1080), the current density is positive when it flows from the electrode into the electrolyte solution. The current densities are positive at the anodes and negative at the cathodes.

The reaction at the cathode electrode is:

$$\frac{1}{2}O_{2,cathode} + 2e^- \Leftrightarrow O^{2-} \tag{21.127}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

or

$$S_{O_2} = -\frac{-\left(\frac{1}{2}\right)\left(-i\right)}{2F} = -\frac{i}{4F} \tag{21.128}$$

$$S_{O}^{2-} = -\frac{\left(-i\right)}{2F} = -\frac{i}{2F} \tag{21.129}$$

The reaction at the anode electrode is:

$$H_2 + O^{2-} \Leftrightarrow H_2O + 2e^- \tag{21.130}$$

$$S_{H_2} = -\frac{i}{2F} \tag{21.131}$$

$$S_{O}^{2-} = -\frac{i}{2F} \tag{21.132}$$

$$S_{H_2O} = -\frac{-\left(1\right)\left(i\right)}{2F} = \frac{i}{2F} \tag{21.133}$$

Note that the total enthalpy includes the formation enthalpy for each species only when the volumetric reactions are enabled. Once the volumetric reactions are enabled, the SOFC model solver can correctly compute the reaction heat.

21.3.5.2. Modeling CO Electrochemistry

In practice, if carbon monoxide (CO) is present in the anode fuel stream, it may be oxidized to generate carbon dioxide (CO_2). This effect is modeled by introducing a $H_2/\ CO$ split factor as follows:

$$\alpha = \frac{x_{H_2}}{x_{H_2} + x_{CO}} \tag{21.134}$$

where x_{H_2} amd x_{CO} are species mole fractions of H_2 and CO, respectively.

The molar source terms for each anode-side species are:

$$S_{H_2} = -\frac{\alpha}{2F} i \tag{21.135}$$

$$S_{CO} = -\frac{1-\alpha}{2F} i \tag{21.136}$$

$$S_{CO_2} = \frac{1-\alpha}{2F} i \tag{21.137}$$

$$S_{H_2O} = \frac{\alpha}{2F} i \tag{21.138}$$

By default, Equation 21.134 (p. 933) is used to compute the $H_2/\ CO$ split factor, however, you can define your own split factor in the user-defined function called `h2_co_split_func()` (see User-Accessible Functions for the Solid Oxide Fuel Cell With Unresolved Electrolyte Model).

21.3.5.3. Modeling Electrolysis

Since electrolysis is a reversed fuel cell process, the SOFC module can also be used to model the electrolysis. In this case, power is supplied to an electrolyzer to convert water vapor into hydrogen and oxygen. Water vapor is fed through the anode electrodes to the active electrolyte region, and the following electrochemical reactions take place:

$$H_2O + 2e^- \ \Leftrightarrow \ H_2 + O^{2-} \ \ \left(anode\ \ TPB\right) \tag{21.139}$$

$$O^{2-} \ \Leftrightarrow \ \frac{1}{2}O_2 + 2e^- \ \ \left(cathode\ \ TPB\right) \tag{21.140}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Note that, conventionally, the negative voltage is supplied to the cathode side in power consuming devices such as electrolyzers. Ansys Fluent adopts the inverse notation where the negative voltage is supplied to the anode side, while cathode remains positively charged. The main reason for this discrepancy is that the same infrastructure is used for both the electrolysis and fuel cells models. Usage of the terms "anode" and "cathode" in this manual and in the GUI should be interpreted according to conventions for power-supplying devices.

In electrolysis, the activation overpotentials have the opposite sign of what is used in fuel cells. This means that the cell voltage is higher than the open circuit voltage, since power is added to overcome the activation overpotentials. The ionic conductivity in the electrolyte is a function of temperature, as in the case of SOFC. The molar reaction rates for hydrogen, oxygen and water vapor are computed based upon the current density i:

$$H_2: \quad +\frac{i}{2F}$$
$$O_2: \quad +\frac{i}{4F}$$
$$H_2O: \quad -\frac{i}{2F}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Chapter 22: Modeling Magnetohydrodynamics

This chapter presents an overview of the theory and the governing equations for the mathematical models used in Ansys Fluent to predict flow in an electromagnetic field.

22.1. Introduction

22.2. Magnetic Induction Method

22.3. Electric Potential Method

22.1. Introduction

The coupling between the fluid flow field and the magnetic field can be understood on the basis of two fundamental effects: the induction of electric current due to the movement of conducting material in a magnetic field, and the effect of Lorentz force as the result of electric current and magnetic field interaction. In general, the induced electric current and the Lorentz force tend to oppose the mechanisms that create them. Movements that lead to electromagnetic induction are therefore systematically braked by the resulting Lorentz force. Electric induction can also occur in the presence of a time-varying magnetic field. The effect is the stirring of fluid movement by the Lorentz force.

Electromagnetic fields are described by Maxwell's equations:

$$\nabla \cdot \vec{B} = 0 \tag{22.1}$$

$$\nabla \times \vec{E} = -\frac{\partial \vec{B}}{\partial t} \tag{22.2}$$

$$\nabla \cdot \vec{D} = q \tag{22.3}$$

$$\nabla \times \vec{H} = \vec{j} + \frac{\partial \vec{D}}{\partial t} \tag{22.4}$$

where $\vec{B}$ (Tesla) and $\vec{E}$ (V/m) are the magnetic and electric fields, respectively, and $\vec{H}$ and $\vec{D}$ are the induction fields for the magnetic and electric fields, respectively. q ($\mathrm{C/m^3}$) is the electric charge density, and $\vec{j}$ ($\mathrm{A/m^2}$) is the electric current density vector.

The induction fields $\vec{H}$ and $\vec{D}$ are defined as:

$$\vec{H} = \frac{1}{\mu}\vec{B} \tag{22.5}$$

$$\vec{D} = \varepsilon \vec{E} \tag{22.6}$$

where μ and ε are the magnetic permeability and the electric permittivity, respectively. For sufficiently conducting media such as liquid metals, the electric charge density q and the displacement current $\frac{\partial \vec{D}}{\partial t}$ are customarily neglected [456] (p. 1079).

In studying the interaction between flow field and electromagnetic field, it is critical to know the current density $\vec{j}$ due to induction. Generally, two approaches may be used to evaluate the current density.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

One is through the solution of a magnetic induction equation; the other is through solving an electric potential equation.

22.2. Magnetic Induction Method

In the first approach, the magnetic induction equation is derived from Ohm's law and Maxwell's equation. The equation provides the coupling between the flow field and the magnetic field.

In general, Ohm's law that defines the current density is given by:

$$\vec{j}=\sigma\vec{E} \tag{22.7}$$

where σ is the electrical conductivity of the media. For fluid velocity field $\vec{U}$ in a magnetic field $\vec{B}$, Ohm's law takes the form:

$$\vec{j}=\sigma\left(\vec{E}+\vec{U}\times\vec{B}\right) \tag{22.8}$$

From Ohm's law and Maxwell's equation, the induction equation can be derived as:

$$\frac{\partial\vec{B}}{\partial t}+\left(\vec{U}\cdot\nabla\right)\vec{B}=\frac{1}{\mu\sigma}\nabla^{2}\vec{B}+\left(\vec{B}\cdot\nabla\right)\vec{U} \tag{22.9}$$

From the solved magnetic field $\vec{B}$, the current density $\vec{j}$ can be calculated using Ampere's relation as:

$$\vec{j}=\frac{1}{\mu}\nabla\times\vec{B} \tag{22.10}$$

Generally, the magnetic field $\vec{B}$ in a MHD problem can be decomposed into the externally imposed field $\vec{B_0}$ and the induced field $\vec{b}$ due to fluid motion. Only the induced field $\vec{b}$ must be solved.

From Maxwell's equations, the imposed field $\vec{B_0}$ satisfies the following equation:

$$\nabla^{2}\vec{B_0}-\mu\sigma'\frac{\partial\vec{B_0}}{\partial t}=0 \tag{22.11}$$

where σ' is the electrical conductivity of the media in which field $\vec{B_0}$ is generated. Two cases need to be considered.

For more information, see the following sections:

22.2.1. Case 1: Externally Imposed Magnetic Field Generated in Non-conducting Media

22.2.2. Case 2: Externally Imposed Magnetic Field Generated in Conducting Media

22.2.1. Case 1: Externally Imposed Magnetic Field Generated in Non-conducting Media

In this case the imposed field $\vec{B_0}$ satisfies the following conditions:

$$\nabla\times\vec{B_0}=0 \tag{22.12}$$

$$\nabla^{2}\vec{B_0}=0 \tag{22.13}$$

With $\vec{B}=\vec{B_0}+\vec{b}$, the induction equation (Equation 22.9 (p. 936)) can be written as:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$\frac{\partial \vec{b}}{\partial t}+\left(\vec{U}\cdot\nabla\right)\vec{b}=\frac{1}{\mu\sigma}\nabla^{2}\vec{b}+\left(\left(\vec{B}_{0}+\vec{b}\right)\cdot\nabla\right)\vec{U}-\left(\vec{U}\cdot\nabla\right)\vec{B}_{0}-\frac{\partial \vec{B}_{0}}{\partial t} \tag{22.14}$$

The current density is given by:

$$\vec{j}=\frac{1}{\mu}\nabla\times\vec{b} \tag{22.15}$$

22.2.2. Case 2: Externally Imposed Magnetic Field Generated in Conducting Media

In this case the conditions given in Equation 22.12 (p. 936) and Equation 22.13 (p. 936) are not true. Assuming that the electrical conductivity of the media in which field $\vec{B}_{0}$ is generated is the same as that of the flow, that is $\sigma'=\sigma$, from Equation 22.9 (p. 936) and Equation 22.11 (p. 936) the induction equation can be written as:

$$\frac{\partial \vec{b}}{\partial t}+\left(\vec{U}\cdot\nabla\right)\vec{b}=\frac{1}{\mu\sigma}\nabla^{2}\vec{b}+\left(\left(\vec{B}_{0}+\vec{b}\right)\cdot\nabla\right)\vec{U}-\left(\vec{U}\cdot\nabla\right)\vec{B}_{0} \tag{22.16}$$

and the current density is given by:

$$\vec{j}=\frac{1}{\mu}\nabla\times\left(\vec{B}_{0}+\vec{b}\right) \tag{22.17}$$

For the induction equation Equation 22.14 (p. 937) or Equation 22.16 (p. 937), the boundary conditions for the induced field are given by:

$$\vec{b}=\left\{b_{n}\quad b_{t1}\quad b_{t2}\right\}^{T}=\vec{b}^{*} \tag{22.18}$$

where the subscripts denote the normal and tangential components of the field and $\vec{b}^{*}$ is specified by you. For an electrically insulating boundary, as $j_{n}=0$ at the boundary, from Ampere's relation one has $b_{t1}=b_{t2}=0$ at the boundary.

22.3. Electric Potential Method

The second approach for the current density is to solve the electric potential equation and calculate the current density using Ohm's law. In general, the electric field $\vec{E}$ can be expressed as:

$$\vec{E}=-\nabla\varphi-\frac{\partial \vec{A}}{\partial t} \tag{22.19}$$

where φ and $\vec{A}$ are the scalar potential and the vector potential, respectively. For a static field and assuming $\vec{b}<<\vec{B}_{0}$, Ohm's law given in Equation 22.8 (p. 936) can be written as:

$$\vec{j}=\sigma\left(-\nabla\varphi+\left(\vec{U}\times\vec{B}_{0}\right)\right) \tag{22.20}$$

For sufficiently conducting media, the principle of conservation of electric charge gives:

$$\nabla\cdot\vec{j}=0 \tag{22.21}$$

The electric potential equation is therefore given by:

$$\nabla^{2}\varphi=\nabla\cdot\left(\vec{U}\times\vec{B}_{0}\right) \tag{22.22}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

The boundary condition for the electric potential φ is given by:

$$\frac{\partial \varphi}{\partial n} = \left(\vec{U} \times \vec{B}_0\right)_{boundary} \cdot \vec{n} \tag{22.23}$$

for an insulating boundary, where $\vec{n}$ is the unit vector normal to the boundary, and

$$\varphi = \varphi_0 \tag{22.24}$$

for a conducting boundary, where φ_0 is the specified potential at the boundary. The current density can then be calculated from Equation 22.20 (p. 937).

With the knowledge of the induced electric current, the MHD coupling is achieved by introducing additional source terms to the fluid momentum equation and energy equation. For the fluid momentum equation, the additional source term is the Lorentz force given by:

$$\vec{F} = \vec{j} \times \vec{B} \tag{22.25}$$

which has units of $\mathrm{N/m^3}$ in the SI system. For the energy equation, the additional source term is the Joule heating rate given by:

$$Q = \frac{1}{\sigma} \vec{j} \cdot \vec{j} \tag{22.26}$$

which has units of $\mathrm{W/m^3}$.

For charged particles in an electromagnetic field, the Lorentz force acting on the particle is given by:

$$\vec{F}_p = q\left(\vec{E} + \vec{v}_p \times \vec{B}\right) \tag{22.27}$$

where q is the particle charge density ($\mathrm{Coulomb/m^3}$) and v_p is the particle velocity. The force $\vec{F}_p$ has units of $\mathrm{N/m^3}$.

For multiphase flows, assuming that the electric surface current at the interface between phases can be ignored, the electric conductivity for the mixture is given by:

$$\sigma_m = \sum_i \sigma_i v_i \tag{22.28}$$

where σ_i and v_i are respectively the electric conductivity and volume fraction of phase i. σ_m is used in solving the induction equations.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Chapter 23: Modeling Continuous Fibers

This chapter presents an overview of the theory and the governing equations for the mathematical model and Ansys Fluent's capabilities to predict melt and dry spinning processes.

23.1. Introduction

23.2. Governing Equations of Fiber Flow

23.3. Discretization of the Fiber Equations

23.4. Numerical Solution Algorithm of Fiber Equations

23.5. Residuals of Fiber Equations

23.6. Coupling Between Fibers and the Surrounding Fluid

23.7. Fiber Grid Generation

23.8. Correlations for Momentum, Heat and Mass Transfer

23.9. Fiber Properties

23.10. Solution Strategies

23.1. Introduction

Ansys Fluent's Continuous Fiber Model uses a one-dimensional approach used to predict the flow in fibers and to predict the flow field in the spinning chamber.

In melt spinning, where the extruded molten polymer is sent through the nozzles into the spinning chamber, the velocity of the liquid jet increases due to gravity and the tensile force, which is applied at the take-up point of the fibers. The conservation of mass leads to a decrease in the cross-section of the jet up to the final diameter. The molten polymer is cooled by an air stream until the solidification temperature is reached.

In dry and melt spinning, production of hundreds or thousands of fibers in a spinning chamber leads to strong interaction between the fibers and the surrounding gas, requiring a coupled calculation procedure for the fibers and the fluid flow in the spinning chamber.

23.2. Governing Equations of Fiber Flow

Mass conservation of a fiber element is written as

$$\frac{d}{dz}\left(\rho_f \vec{u}_f \vec{A}_f Y_s\right) = -\pi d_f \dot{m}_s'' \tag{23.1}$$

where

ρ_f = fiber density

$\vec{u}_f$ = fiber velocity vector

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$\vec{A}_f$ = surface area vector of the fiber surface parallel to the flow direction

Y_s = mass fraction of the solvent s in the fiber

$\dot{m}_s''$ = evaporated mass flow rate of the solvent s

d_f = fiber diameter

z = coordinate along the fiber

$\dot{m}_s''$ is calculated using a film theory.

$$\dot{m}_s''=M_s c\beta \ln\left(\frac{1-\psi_{s,g}}{1-\psi_{s,I}}\right) \tag{23.2}$$

where

β = mass transfer coefficient estimated from an appropriate correlation (see Correlations for Momentum, Heat and Mass Transfer (p. 948))

M_s = solvent's molecular weight

c = molar concentration of the surrounding gas

$\psi_{s,g}$ = mole fraction of the solvent vapor in the surrounding gas

At the fiber surface, the mole fraction of the solvent in the gas $\psi_{s,I}$ is related to the solvent mass fraction in the fiber Y_s by the vapor-liquid equilibrium equation given by Flory [185] (p. 1063),

$$\psi_{s,I}=\frac{p_{s,vap}}{p} Y_s e^{1-Y_s+\chi\left(1-Y_s\right)^2} \tag{23.3}$$

where

χ = Flory-Huggins parameter

p = absolute pressure in the surrounding flow

$p_{s,vap}$ = saturation vapor pressure of the solvent

These equations are used only when dry spun fibers have been selected.

The formation of fibers is based on tensile forces in the fiber that are applied at the take-up point and result in the drawing and elongation of the fiber.

A force balance for a differential fiber element gives the equation of change of momentum in the fiber.

$$\frac{d\left(\rho_f u_f u_f \vec{A}_f\right)}{dz}=\frac{d\vec{F}}{dz}+\vec{F}_{friction}+\vec{F}_{gravitation} \tag{23.4}$$

The tensile force in the fiber changes due to acceleration of the fiber, friction force with the surrounding gas, and the gravitational forces.

The friction force with the surrounding gas is computed by

$$\vec{F}_{friction}=\frac{1}{2}\rho c_{f,ax}\pi d_f\left|\vec{u}_f-\vec{u}_{par}\right|\left(\vec{u}_f-\vec{u}_{par}\right) \tag{23.5}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

where

ρ = gas density

$c_{f,ax}$ = axial friction factor parallel to the fiber

$\vec{u}_{par}$ = gas velocity parallel to the fiber

The gravitational force is computed from

$$\vec{F}_{gravitation}=\rho_f \frac{\pi}{4} d_f^2 \vec{g}\cdot\vec{n}_f \tag{23.6}$$

where $\vec{n}_f$ is the direction vector of the fiber element.

The tensile force $\vec{F}$ is related to the components of the stress tensor by

$$\vec{F}=\vec{A}_f\left(\tau_{zz}-\tau_{rr}\right) \tag{23.7}$$

Neglecting visco-elastic effects and assuming Newtonian flow one can obtain

$$\tau_{zz} = 2\eta_f \frac{du_f}{dz} \tag{23.8}$$

$$\tau_{rr} = -\eta_f \frac{du_f}{dz} \tag{23.9}$$

leading to

$$\vec{F}=3\vec{A}_f\eta_f \frac{du_f}{dz} \tag{23.10}$$

The elongational viscosity is estimated by multiplying the zero shear viscosity η_f by three.

The transport of enthalpy in and to a differential fiber element is balanced to calculate the fiber temperature along the spinning line.

$$\begin{aligned} \frac{d}{dz}\left(\rho_f \vec{u}_f\cdot\vec{A}_f h_f\right) &= \frac{d}{dz}\left(\lambda_f \vec{A}_f \frac{dT_f}{dz}\right)+\pi d_f\left(\alpha\left(T-T_f\right)-\dot{m}_s'' h_{s,v}\right) \\ &+ \dot{Q}_{viscous\ heating}+\dot{Q}_{radiation,abs}-\dot{Q}_{radiation,emission} \end{aligned} \tag{23.11}$$

where

h_f = fiber enthalpy

λ_f = fiber thermal conductivity

T_f = fiber temperature

$h_{s,v}$ = enthalpy of the solvent vapor

α = heat transfer coefficient

In the case of a melt spinning process, $\dot{m}_s''$ is zero because there is no mass transfer. The term for heat generation due to viscous heating is derived from the fluid mechanics of cylindrical flow to be

$$\dot{Q}_{viscous\ heating}=\frac{\pi}{4} d_f^2\left(4\left(\frac{du_f}{dz}\right)^2-\frac{2}{3}\dot{m}_s'''^2\right) \tag{23.12}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Radiation heat exchange is considered by the last two terms

$$\dot{Q}_{radiation,abs} = d_f \varepsilon_f G \tag{23.13}$$

$$\dot{Q}_{radiation,emission} = \pi d_f \varepsilon_f \sigma T_f^4 \tag{23.14}$$

where

G = thermal irradiation

ε_f = fiber's emissivity

σ = Boltzman constant

The fiber enthalpy h_f is related to the fiber temperature T_f as follows

$$h_f = \int_{T_{ref}}^{T_f} \left(\left(1 - Y_s\right) C_{p_p} + Y_s C_{p_s} \right) dT \tag{23.15}$$

It uses C_{p_p} the specific heat capacity of the polymer and C_{p_s} the specific heat capacity of the solvent in the fiber.

The enthalpy of the solvent vapor at a given temperature T_v depends on the heat of vaporization Δh_s, given at the vaporization temperature T_{vap}, and is computed from

$$h_{s,v} = \int_{T_{ref}}^{T_{vap}} C_{p_{s,l}} dT + \Delta h_s \Big|_{T_{vap}} + \int_{T_{vap}}^{T_v} C_{p_{s,v}} dT \tag{23.16}$$

where

$C_{p_{s,l}}$= specific heat capacity of the solvent liquid

$C_{p_{s,v}}$ = specific heat capacity of the solvent vapor

23.3. Discretization of the Fiber Equations

The governing Equation 23.1 (p. 939), Equation 23.4 (p. 940), and Equation 23.11 (p. 941) have the form of convection diffusion equations and are discretized using a finite volume scheme of Patankar [505] (p. 1082).

The convective terms in Equation 23.1 (p. 939), Equation 23.4 (p. 940), and Equation 23.11 (p. 941) are discretized using first order upwinding, central differencing or the DISC scheme [547] (p. 1084). While the first two schemes are well described in the Ansys Fluent User's Guide, the reader is referred to [547] (p. 1084) to learn more about the DISC scheme. It is the recommended scheme because it provides outstanding numerical stability combined with second order accuracy. The diffusion terms are discretized with second order accuracy. All other terms are treated as source terms and linearized according to Patankar [505] (p. 1082).

For more information, see the following section:

23.3.1. Under-Relaxation

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

23.3.1. Under-Relaxation

Because the fiber equations are nonlinear, the change of the solution variable ϕ has to be controlled. This is achieved by under-relaxation. The new value of ϕ in each cell depends to some degree upon the old value ϕ_{old} and the change in ϕ, $\Delta\phi$. It is computed for a given under-relaxation factor α as follows:

$$\phi=\phi_{old}+\alpha\Delta\phi \tag{23.17}$$

23.4. Numerical Solution Algorithm of Fiber Equations

Each governing differential equation is discretized into a set of algebraic equations that are solved using the tri-diagonal matrix algorithm. All differential equations for conservation of mass, momentum, energy, and (when appropriate) for solvent in the fiber are solved sequentially (that is, segregated from one another). Because the governing equations are coupled and nonlinear, several iterations have to be performed to obtain a converged solution. The solution process consists of several steps outlined below:

1. The fiber properties are updated based on the initialized or the current solution.
2. The friction factors for momentum exchange between the fibers and surrounding fluid are computed based on current values of fiber and fluid velocities.
3. The fiber momentum equation is solved and current values of the mass fluxes in the fiber are used.
4. The heat transfer coefficients are computed using Reynolds numbers from the beginning of the iteration loop.
5. The fiber energy equation is solved.
6. In the case of dry spun fibers, the equation for the mass fraction of the solvent is solved. First, the mass transfer coefficient is updated. The evaporated (condensed) mass is computed based on the vapor liquid equilibrium at the beginning of the iteration loop. Finally, the governing equation is solved.
7. The mass fluxes and the diameter of the fiber cells are updated.
8. A check for convergence of the equation set is made.

These steps are continued until the convergence criteria are met for all equations of the considered fiber or until the number of iterations exceed the given limit.

This solution algorithm is applied to all defined fibers.

23.5. Residuals of Fiber Equations

The solution algorithm for the fiber equations requires a means for checking convergence of the solution. In the fiber model a simple residual is used for this purpose.

The conservation equation for a general variable ϕ at a cell P can be written as

$$a_P\phi_P=\sum_{nb}a_{nb}\phi_{nb}+S_c+S_P\phi_P \tag{23.18}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

where

a_P = center coefficient

a_{nb} = influence coefficients for the neighboring cells

S_c = constant part of the source term

S_P = linear part of the source term

The residual R^{ϕ} computed by the fiber model is the imbalance in Equation 23.18 (p. 943) summed over all fiber cells.

$$R^{\phi} = \sum_{fibercells} \left| \sum_{nb} a_{nb}\phi_{nb} + S_c + S_P\phi_P - a_P\phi_P \right| \tag{23.19}$$

This is called the absolute residual. Relative residuals are defined as the change of the absolute residuals between two subsequent iterations divided by the absolute residual.

$$\hat{R}^{\phi} = \frac{R^{\phi}_{iterationN} - R^{\phi}_{iterationN-1}}{R^{\phi}_{iterationN}} \tag{23.20}$$

23.6. Coupling Between Fibers and the Surrounding Fluid

The fiber model accounts for all mass, momentum, and heat gained or lost by the fibers and can be incorporated into the subsequent calculations of the surrounding fluid performed by Ansys Fluent. This means that you can incorporate the effects of the fibers on the surrounding fluid, while the surrounding fluid always impacts the fiber flow. This two-way coupling is achieved by alternatively solving the fiber and the surrounding fluid flow equations until both phases have converged. The interphase exchange of heat, mass, and momentum from the fibers to the surrounding fluid is depicted qualitatively in Figure 23.1: Fiber Grid Penetrating Grid of the Gas Flow (p. 945).

For more information, see the following sections:

23.6.1. Momentum Exchange

23.6.2. Mass Exchange

23.6.3. Heat Exchange

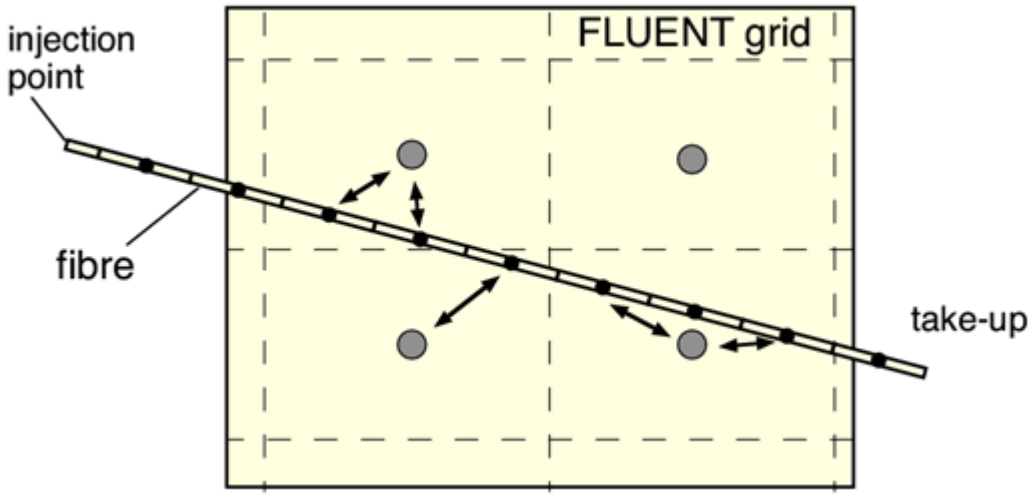
23.6.4. Radiation Exchange

23.6.5. Under-Relaxation of the Fiber Exchange Terms

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

23.6.1. Momentum Exchange

Figure 23.1: Fiber Grid Penetrating Grid of the Gas Flow



Momentum transfer from the fibers to the surrounding fluid is computed in Ansys Fluent by considering the change of momentum of the fiber as it crosses each control volume in the Ansys Fluent model. It is computed as

$$\vec{F}_c = \sum_{fibers} \left(\frac{\rho}{2} \pi d_f l_{f,c} \left(c_{f,par} \left(\vec{u}_f - \vec{u}_{par} \right)^2 - c_{f,lat} \left(\vec{u}_f - \vec{u}_{lat} \right)^2 \right) + \vec{u}_f \pi d_f l_{f,c} \dot{m}_s'' \right) \tag{23.21}$$

where

ρ = density of the fluid

d_f = diameter of the fiber

$l_{f,c}$ = length of the fiber f in cell c

$\vec{u}_f$ = velocity of the fiber

$\vec{u}_{par}$ = velocity of the fluid, parallel to the fiber

$\vec{u}_{lat}$ = velocity of the fluid, lateral to the fiber

$c_{f,par}$ = drag coefficient parallel to the fiber

$c_{f,lat}$ = drag coefficient lateral to the fiber

$\dot{m}_s''$ = evaporated mass flow rate of the solvent in the fiber

This momentum exchange appears as a momentum source in the surrounding fluid momentum balance and is taken into account during every continuous phase computation. It can be reported as described in Exchange Terms of Fibers.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

23.6.2. Mass Exchange

The mass transfer of evaporating solvent from the fibers in dry spinning applications is computed in Ansys Fluent by balancing the evaporated mass of solvent in every fiber cell volume. It is computed as

$$M_c = \sum_{fibers} \pi d_f l_{f,c} \dot{m}_s'' \tag{23.22}$$

The mass exchange appears as a mass source in the continuity equation of the surrounding fluid as well as a source of chemical species of the solvent vapor defined by you. It is included in every subsequent calculation of the continuous phase flow field and is reported as described in Exchange Terms of Fibers.

23.6.3. Heat Exchange

The heat transfer from the fibers to the surrounding flow is computed in Ansys Fluent by balancing the change of the fiber energy as it crosses each control volume in the Ansys Fluent model. It considers sensible as well as latent heat transfer due to evaporation of solvent in dry spinning applications and is computed as

$$Q_c = \sum_{fibers} \left(\pi d_f l_{f,c} \left(\alpha \left(T_f - T \right) + \dot{m}_s'' \left(\frac{1}{2} \vec{u}_f^2 + \int_{T_{ref}}^{T_f} C_{p_{s,v}} dT \right) \right) + \vec{u}_f \vec{F}_f \right) \tag{23.23}$$

where

α = heat transfer coefficient

T_{ref} = reference temperature

$C_{p_{s,v}}$ = specific heat capacity of solvent vapor

$\vec{F}_f$ = momentum exchange of fiber f

This heat exchange appears as a source in the surrounding fluid energy balance and is taken into account during every continuous phase computation. It can be reported as described in Exchange Terms of Fibers.

23.6.4. Radiation Exchange

The fibers participate in radiation exchange by absorbing irradiation energy from the surrounding flow irradiation and by emitting irradiation at the fiber temperature. This effect on the irradiation of the surrounding flow is considered by computing the absorbed and emitted energy of the fibers in each cell as

$$G_{abs} = \sum_{fibers} \frac{\varepsilon_f}{4} A_f G \tag{23.24}$$

$$G_{emiss} = \sum_{fibers} \frac{\varepsilon_f}{4} \frac{A_f}{\pi} \sigma T_f^4 \tag{23.25}$$

where

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

A_f = fiber surface area

σ = Boltzman constant

G = irradiation of the surrounding flow

ε_f = emissivity of the fiber

Important:

The transfer of the fiber radiation energy to the surrounding flow is only considered when the single-band Discrete-Ordinate Model is used in the Ansys Fluent flow model. While radiation effects on the fibers are taken into account when the P1 model is used, there is no two-way coupling for this model.

23.6.5. Under-Relaxation of the Fiber Exchange Terms

Note that the exchange of momentum, heat, and mass from the fibers is under-relaxed during the calculation, so that

$$\vec{F}_{c,new} = \vec{F}_{c,old} + \alpha\left(\vec{F}_{c,calculated} - \vec{F}_{c,old}\right) \tag{23.26}$$

$$M_{c,new} = M_{c,old} + \alpha\left(M_{c,calculated} - M_{c,old}\right) \tag{23.27}$$

$$Q_{c,new} = Q_{c,old} + \alpha\left(Q_{c,calculated} - Q_{c,old}\right) \tag{23.28}$$

where α is the under-relaxation factor for fibers that you can set in the **Solution Controls** task page. The default value for α is 0.5. This value may be reduced in order to improve the stability of coupled calculations. Note that the value of α does not influence the predictions obtained in the final converged solution.

23.7. Fiber Grid Generation

The fibers penetrate the grid of the surrounding flow field arbitrarily. Both grids are treated distinctly from each other (see Figure 23.1: Fiber Grid Penetrating Grid of the Gas Flow (p. 945)).

The fiber grids are generated by defining fiber injections. The fibers are considered to be straight lines between injection points and a take-up point. Each fiber is divided into a number of volume cells.

For the grid generation, the following grid types are available:

equidistant

All cells of the fiber have the same length.

one-sided

The cells are graded near the injection point of the fiber and change their size according to a specified growth factor.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

two-sided

In addition to the injection point the cells can also be graded at the take-up point by specifying a second growth factor at the end of the fiber.

three-sided

The second point where the fiber cells are graded at the end can be moved to a local refinement point laying between injection and take-up point. This generates fibers with a mesh graded at the injection point and at the local refinement point.

All grid types, except the **equidistant** grid type, require the specification of a growth factor R, which is the ratio of two subsequent fiber grid cells. It refines the mesh for values larger than 1 and coarsens the mesh for values smaller than 1.

If a finite fiber volume cell spans across several Ansys Fluent grid cells, a weighted average is used to estimate the corresponding variables of the surrounding flow. This averaging procedure considers the intersection point of each fiber volume cell with the boundaries of the Ansys Fluent grid cell.

When computing the source terms in an Ansys Fluent grid cell, only the part of each fiber volume cell that is inside the Ansys Fluent grid cell is taken into account. This provides a proper computation of Ansys Fluent fiber interactions even in hanging node adapted grids.

Important:

If the grid is adapted, the data structures that include information about neighbor cells are not updated automatically. For this you have to reinitialize all fibers to start a new search of the neighboring Ansys Fluent grid cells.

23.8. Correlations for Momentum, Heat and Mass Transfer

The fiber model makes use of correlations to compute transfer of momentum, heat, and mass to the fibers. The fibers are subject to several physical effects that can be considered based on experimental correlations. Some of these effects may be lateral or longitudinal oscillations due to the applied take-up system, or to gas flow turbulence in the spinning chamber caused by the gas supply system of the spinning chamber. You can choose to specify a constant for the drag, heat transfer or mass transfer coefficients or use one of the Fluent-provided methods (for example, `kase-matsuo`) to compute the coefficients. These methods are described below. Alternatively, you can specify a custom drag, heat transfer, or mass transfer coefficient using a user-defined function (UDF). See User-Defined Functions (UDFs) for the Continuous Fiber Model for details on how to define and use UDFs in the fiber model.

For more information, see the following sections:

23.8.1. Drag Coefficient

23.8.2. Heat Transfer Coefficient

23.8.3. Mass Transfer Coefficient

23.8.1. Drag Coefficient

The following options for drag coefficients are available in the fiber model to compute the drag due to flow moving parallel to the fibers:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

const-drag

A constant value for the drag can be specified.

kase-matsuo

A drag coefficient using the model taken from Kase and Matsuo [290] (p. 1069), see Equation 23.29 (p. 950).

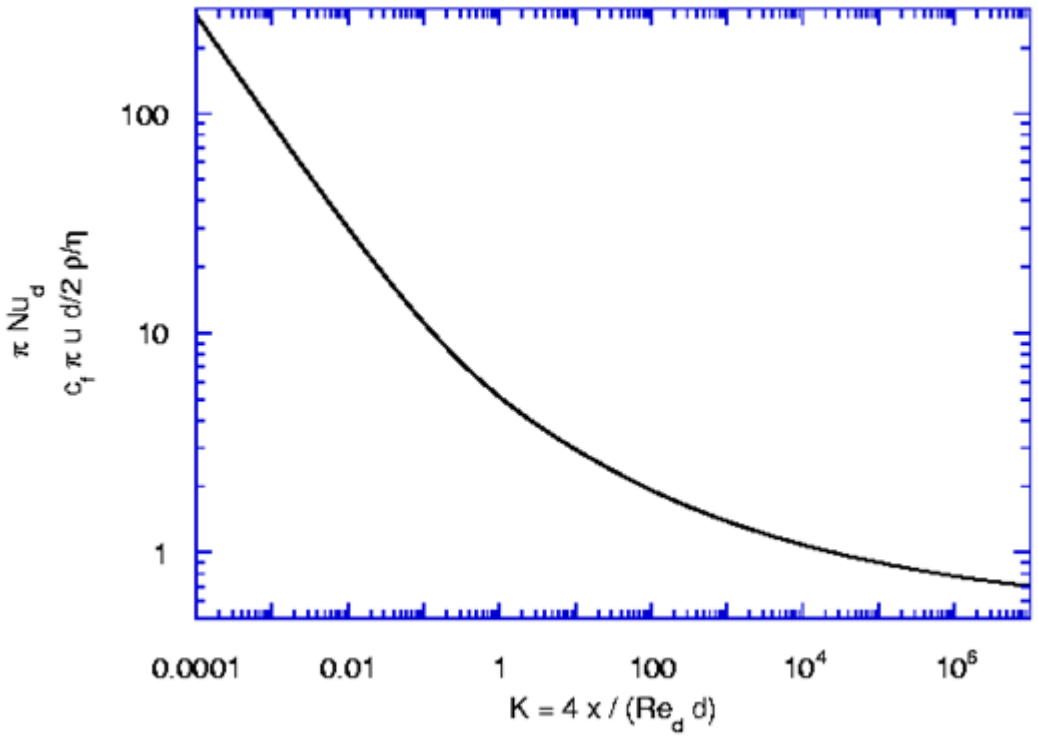
gampert

A drag coefficient using the model from Gampert [195] (p. 1064).

Gampert provided analytical and numerical solutions for laminar axisymmetric flow of a moving cylinder in stationary air including strong curvature effects in the boundary layer, [195] (p. 1064). The drag coefficient and the Nusselt number are shown as dimensionless groups in Figure 23.2: Dimensionless Groups of Drag Coefficient and Nusselt Number (p. 949).

Note that the curvature k is defined as the abscissa in Figure 23.2: Dimensionless Groups of Drag Coefficient and Nusselt Number (p. 949). This correlation is recommended in laminar flows.

Figure 23.2: Dimensionless Groups of Drag Coefficient and Nusselt Number



user-defined

A drag coefficient that you specify in a user-defined function (UDF). See User-Defined Functions (UDFs) for the Continuous Fiber Model for more information on using UDFs in the fiber model.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$c_{f,par} = \frac{1.24}{Re_d^{0.81}} \tag{23.29}$$

In Figure 23.2: Dimensionless Groups of Drag Coefficient and Nusselt Number (p. 949) and Equation 23.29 (p. 950), the Reynolds number is computed based on the relative velocity of the surrounding flow parallel to the fibers $Re_d = \frac{\rho d \left(u_f - u_{par}\right)}{\eta}$.

Lateral drag due to flow of the surrounding fluid perpendicular to the fibers is computed by a correlation from Schlichting [576] (p. 1086)

$$c_{f,lat} = 10^{\left(a_1 + a_2 \log Re_{d,lat} + a_3 \log^2 Re_{d,lat}\right)} \tag{23.30}$$

In Equation 23.30 (p. 950) the Reynolds number is computed based on the relative velocity of the surrounding flow perpendicular to the fibers $Re_{d,lat} = \frac{\rho d u_{lat}}{\eta}$.

23.8.2. Heat Transfer Coefficient

The following options for heat transfer coefficients are available in the fiber model to compute the exchange of heat between fibers and surrounding flow:

const-htc

A constant value for the heat transfer coefficient in SI units can be specified.

kase-matsuo-1

A heat transfer coefficient based on a model from Kase and Matsuo [290] (p. 1069) that considers pure parallel flow, see Equation 23.31 (p. 950).

$$Nu_d = \frac{\alpha d_f}{\lambda} = 0.42 Re_d^{0.334} \tag{23.31}$$

kase-matsuo-2

A heat transfer coefficient based on a model from Kase and Matsuo [290] (p. 1069) that also considers cross flow, see Equation 23.32 (p. 950). Refer to Momentum Exchange (p. 945) for definitions of the variables below.

$$Nu_d = \frac{\alpha d_f}{\lambda} = 0.42 Re_d^{0.334} \left[1 + \left(\frac{8 u_{lat}}{u_f - u_{par}} \right)^2 \right]^{1/6} \tag{23.32}$$

gampert

A heat transfer coefficient based on a model from Gampert [195] (p. 1064).

Gampert provided analytical and numerical solutions for laminar axisymmetric flow of a moving cylinder in stationary air including strong curvature effects in the boundary layer, [195] (p. 1064). The drag coefficient and the Nusselt number are shown as dimensionless groups in Figure 23.2: Dimensionless Groups of Drag Coefficient and Nusselt Number (p. 949). This correlation is recommended for laminar flows.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

user-defined

A heat transfer coefficient that you specify in a user-defined function (UDF). See User-Defined Functions (UDFs) for the Continuous Fiber Model for more information on using UDFs in the fiber model.

23.8.3. Mass Transfer Coefficient

The mass transfer coefficients are based on the Reynolds analogy applied to the heat transfer coefficients. The following options are available in the fiber model to compute the exchange of mass between fibers and surrounding flow:

const-mtc

If you select this coefficient, you specify the direct transferred mass flow rate in $kg/\left(m^2 s\right)$, rather than the mass transfer coefficient.

kase-matsuo-1

A mass transfer coefficient based on a model from Kase and Matsuo [290] (p. 1069) that considers pure parallel flow, see Equation 23.33 (p. 951).

$$Sh_d = \frac{\beta d_f}{D_s} = 0.42 Re_d^{0.334} \tag{23.33}$$

kase-matsuo-2

A mass transfer coefficient based on a model from Kase and Matsuo [290] (p. 1069) that also considers cross flow, see Equation 23.34 (p. 951).

$$Sh_d = \frac{\beta d_f}{D_s} = 0.42 Re_d^{0.334} \left[1 + \left(\frac{8 u_{lat}}{u_f - u_{par}} \right)^2 \right]^{1/6} \tag{23.34}$$

gampert

A mass transfer coefficient based on a model from Gampert [195] (p. 1064).

Gampert's analytical and numerical solutions for laminar axisymmetric flow of a moving cylinder in stationary air include strong curvature effects in the boundary layer, [195] (p. 1064). The Sherwood number is analogous to the Nusselt number as shown as dimensionless groups in Figure 23.2: Dimensionless Groups of Drag Coefficient and Nusselt Number (p. 949). This correlation is recommended for laminar flows.

user-defined

A mass transfer coefficient that you specify in a user-defined function (UDF). See User-Defined Functions (UDFs) for the Continuous Fiber Model for more information on using UDFs in the fiber model.

Because these correlations are valid only for nearly-zero mass transfer due to the Reynolds analogy, a film theory is used to compute the nonzero mass transfer, see Equation 23.2 (p. 940).

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

23.9. Fiber Properties

Most properties needed for the fibers can be specified in Ansys Fluent's material dialog box except the ones described in this section.

23.9.1. Fiber Viscosity
23.9.2. Vapor-Liquid Equilibrium
23.9.3. Latent Heat of Vaporization
23.9.4. Emissivity

23.9.1. Fiber Viscosity

Fibers are treated as Newtonian fluids. In elongational flow of Newtonian fluids, the elongational viscosity (or sometimes called Trouton viscosity) is related to the zero shear viscosity by a factor of 3. Because this approach is applied to the computation of melt and dry spun fibers, only the zero shear viscosity is described.

23.9.1.1. Melt Spinning

In melt spinning, the fiber is considered to be liquid until its temperature falls below the solidification temperature $T_{f,solid}$. For the liquid state an exponential approach is used, see Equation 23.35 (p. 952).

$$\eta_0 = Ae^{\left(\frac{B}{C+T_f}\right)}\left(\frac{du_f}{dz}\right)^{(N-1)} \tag{23.35}$$

where A, B, C, and N are user-specified constants.

Below this temperature the value given in the material dialog box for the fiber polymer material is used. Typically, a high value like $\eta_0 = 1x10^8 Pas$ is used for the fiber viscosity to simulate a solid fiber. This value may depend on your polymer and the range of viscosity values in your simulation. You can use every profile available in the materials dialog box except UDF's to describe temperature dependency of the viscosity of the solidified fiber.

The fiber model uses a blending interval for the temperature $\Delta T_{f,bl} = T_{f,liquid} - T_{f,solid}$ to provide a smooth transition of the viscosity between liquid and solid state of the fiber. The viscosity in this blending interval is computed as

$$\eta_0 = \frac{\Delta T_{f,bl}}{\frac{T_{f,liquid} - T_f}{\eta_{f,solid}(T_f)} + \frac{T_f - T_{f,solid}}{\eta_{f,liquid}(T_f)}} \tag{23.36}$$

Important:

The chosen values of the blending interval may influence the results. Values for the blending interval should be adapted to the rheological data of the polymer.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

23.9.1.2. Dry Spinning

In dry spinning, the following approach is used to consider the effect of solvent on the zero shear viscosity:

$$\eta_0 = AB^C \left(\frac{du}{dz} \right)^{(N-1)} \left(1 - Y_s \right)^D e^{\left(E/T_f \right)} \tag{23.37}$$

In this equation, B is considered to be the degree of polymerization as it is used by Ohzawa [484] (p. 1081). A, B, C, D, E, and N are user-specified constants.

23.9.2. Vapor-Liquid Equilibrium

In dry spinning applications, the solvent in the fiber is in thermodynamic equilibrium with the solvent vapor in the surrounding fluid. The saturation vapor pressure of the solvent is computed using an Antoine-type equation

$$p_{s,vap} = e^{\left(A + \frac{B}{T_f + C} \right)} \tag{23.38}$$

If you enable the vapor-liquid equilibrium given by Flory [185] (p. 1063), Equation 23.3 (p. 940) is used to compute the mole fraction of the solvent in the surrounding gas.

23.9.3. Latent Heat of Vaporization

The latent heat of vaporization of the solvent Δh_s has to be specified at the vaporization temperature T_{vap}. The temperature dependency of the latent heat is not needed because this is treated automatically by the fiber model. Using Equation 23.16 (p. 942) the following equation can be derived to get the latent heat of vaporization at any temperature $T^\star$:

$$\Delta h_s \left(T^\star \right) = \Delta h_s \left(T_{vap} \right) + \int_{T_{vap}}^{T^\star} \left(C_{p_{s,v}} - C_{p_s} \right) dT \tag{23.39}$$

23.9.4. Emissivity

If the P-1 or the discrete ordinates radiation model is enabled, the emissivity of the fiber has to be specified. The fiber diameter can approach the order of magnitude of the wavelength of the irradiation. In this parameter range, additional effects like diffraction take place in addition to scattering and transmission. Because these properties are not well known for fibers, diffraction, scattering, and transmission are neglected. They have to be included only in the fiber emissivity.

23.10. Solution Strategies

The fiber model solves sets of differential equations. It provides its own numerical algorithms showing their own numerical difficulties. Therefore it is highly recommended to first achieve a numerical solution for the pure fibers without any coupling to the surrounding flows. If the fiber model does not converge for a given flow field of the surrounding fluid it will not converge for changing fields of the surrounding fluid.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

When you start with a fiber simulation, choose the appropriate models needed for the fibers. For the first simulation, disable options like **Include Lateral Drag**, **Fiber Radiation Interaction**, and **Fiber Viscous Heating** to reduce possible interactions.

When specifying the grid of the fibers be sure to refine the grid in the area where large gradients of the velocity appear. This is mainly near the injection point where the fiber is released and near the point of solidification. Because this point is not known a priori, you have to refine the grid during subsequent steps.

If the fiber grid seems to be well suited, you can influence the convergence behavior by starting the iteration with low under-relaxation factors in the **Fiber Solution Controls** dialog box. This may help in most situations where the species and energy equations are strongly coupled (for example, dry spinning applications), or if the solvent has a very high latent heat of vaporization (for example, water).

Important:

Be sure to increase the under-relaxation factor of the momentum equation to 1, when doing a melt spinning case to achieve a converged solution. This should be done after a numerically stable solution has been set up.

When the solution process of the pure fiber equations show a numerically stable behavior, you can increase the complexity of the models by activating viscous heating, or radiation interaction, if such effects are important in their application. After this, you can proceed with a coupled solution by solving the fiber equations and the fluid flow equations alternately. If the solution algorithm of the fluid flow equations diverges, you must investigate the source terms computed by the fiber model, see Exchange Terms of Fibers.

You can damp strong changes of the source terms with a low under-relaxation factor. Another choice is to increase the number of Ansys Fluent iterations between two subsequent fiber computations.

If the coupled solution process converges, you can increase the under-relaxation factor of the source terms and decrease the number of Ansys Fluent iterations between two subsequent fiber computations.

You may also want to consider underrelaxing the fluid flow equations. This helps especially for the energy and species equations.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Chapter 24: Solver Theory

This chapter describes the Ansys Fluent solver theory. Details about the solver algorithms used by Ansys Fluent are provided in Overview of Flow Solvers (p. 955) – Multigrid Method (p. 1003). For more information about using the solver, see Using the Solver in the User's Guide.

24.1. Overview of Flow Solvers

24.2. General Scalar Transport Equation: Discretization and Solution

24.3. Discretization

24.4. Pressure-Based Solver

24.5. Density-Based Solver

24.6. Pseudo Time Method Under-Relaxation

24.7. Multigrid Method

24.8. Hybrid Initialization

24.9. Full Multigrid (FMG) Initialization

24.1. Overview of Flow Solvers

Ansys Fluent allows you to choose one of the two numerical methods:

- pressure-based solver (see Pressure-Based Solver (p. 956))
- density-based solver (see Density-Based Solver (p. 958))

Historically speaking, the pressure-based approach was developed for low-speed incompressible flows, while the density-based approach was mainly used for high-speed compressible flows. However, recently both methods have been extended and reformulated to solve and operate for a wide range of flow conditions beyond their traditional or original intent.

In both methods the velocity field is obtained from the momentum equations. In the density-based approach, the continuity equation is used to obtain the density field while the pressure field is determined from the equation of state.

On the other hand, in the pressure-based approach, the pressure field is extracted by solving a pressure or pressure correction equation which is obtained by manipulating continuity and momentum equations.

Using either method, Ansys Fluent will solve the governing integral equations for the conservation of mass and momentum, and (when appropriate) for energy and other scalars such as turbulence and chemical species. In both cases a control-volume-based technique is used that consists of:

- Division of the domain into discrete control volumes using a computational grid.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

- Integration of the governing equations on the individual control volumes to construct algebraic equations for the discrete dependent variables ("unknowns") such as velocities, pressure, temperature, and conserved scalars.
- Linearization of the discretized equations and solution of the resultant linear equation system to yield updated values of the dependent variables.

The two numerical methods employ a similar discretization process (finite-volume), but the approach used to linearize and solve the discretized equations is different.

The general solution methods are described in Pressure-Based Solver (p. 956) and Density-Based Solver (p. 958). To learn how to apply the solvers, see Choosing the Solver in the User's Guide.

24.1.1. Pressure-Based Solver

The pressure-based solver employs an algorithm which belongs to a general class of methods called the projection method [108] (p. 1059). In the projection method, wherein the constraint of mass conservation (continuity) of the velocity field is achieved by solving a pressure (or pressure correction) equation. The pressure equation is derived from the continuity and the momentum equations in such away that the velocity field, corrected by the pressure, satisfies the continuity. Since the governing equations are nonlinear and coupled to one another, the solution process involves iterations wherein the entire set of governing equations is solved repeatedly until the solution converges.

Two pressure-based solver algorithms are available in Ansys Fluent. A segregated algorithm, and a coupled algorithm. These two approaches are discussed in the sections below.

24.1.1.1. The Pressure-Based Segregated Algorithm

The pressure-based solver uses a solution algorithm where the governing equations are solved sequentially (that is, segregated from one another). Because the governing equations are nonlinear and coupled, the solution loop must be carried out iteratively in order to obtain a converged numerical solution.

In the segregated algorithm, the individual governing equations for the solution variables (for example, u, v, w, p, T, k, ε, and so on) are solved one after another. Each governing equation, while being solved, is "decoupled" or "segregated" from other equations, hence its name. The segregated algorithm is memory-efficient, since the discretized equations need only be stored in the memory one at a time. However, the solution convergence is relatively slow, inasmuch as the equations are solved in a decoupled manner.

With the segregated algorithm, each iteration consists of the steps illustrated in Figure 24.1: Overview of the Pressure-Based Solution Methods (p. 957) and outlined below:

1. Update fluid properties (for example, density, viscosity, specific heat) including turbulent viscosity (diffusivity) based on the current solution.
2. Solve the momentum equations, one after another, using the recently updated values of pressure and face mass fluxes.
3. Solve the pressure correction equation using the recently obtained velocity field and the mass-flux.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

4. Correct face mass fluxes, pressure, and the velocity field using the pressure correction obtained from Step 3.

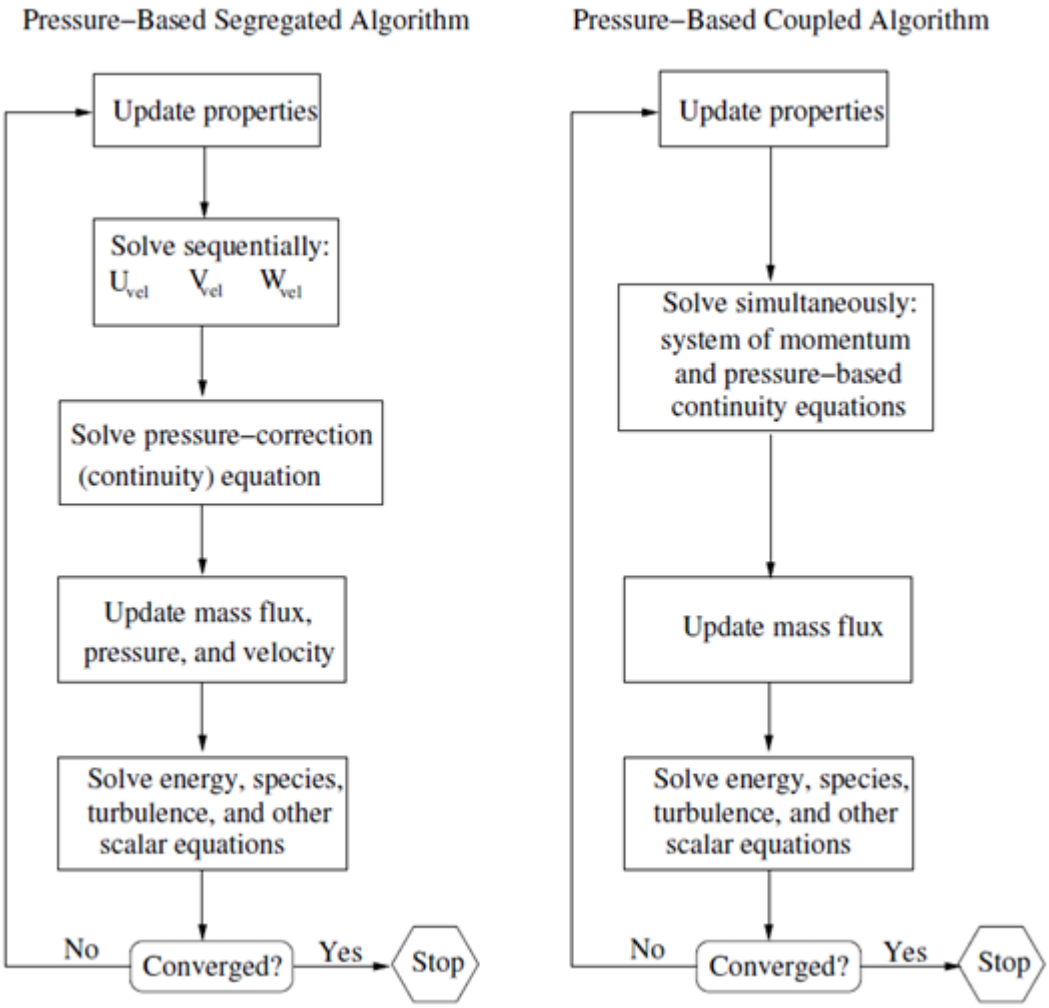
5. Solve the equations for additional scalars, if any, such as turbulent quantities, energy, species, and radiation intensity using the current values of the solution variables.

6. Update the source terms arising from the interactions among different phases (for example, source term for the carrier phase due to discrete particles).

7. Check for the convergence of the equations.

These steps are continued until the convergence criteria are met.

Figure 24.1: Overview of the Pressure-Based Solution Methods



24.1.1.2. The Pressure-Based Coupled Algorithm

Unlike the segregated algorithm described above, the pressure-based coupled algorithm solves a coupled system of equations comprising the momentum equations and the pressure-based continuity equation. Thus, in the coupled algorithm, Steps 2 and 3 in the segregated solution algorithm

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

are replaced by a single step in which the coupled system of equations are solved. The remaining equations are solved in a decoupled fashion as in the segregated algorithm.

Since the momentum and continuity equations are solved in a closely coupled manner, the rate of solution convergence significantly improves when compared to the segregated algorithm. However, the memory requirement increases by 1.5 – 2 times that of the segregated algorithm since the discrete system of all momentum and pressure-based continuity equations must be stored in the memory when solving for the velocity and pressure fields (rather than just a single equation, as is the case with the segregated algorithm).

24.1.2. Density-Based Solver

The density-based solver solves the governing equations of continuity, momentum, and (where appropriate) energy and species transport simultaneously (that is, coupled together). Governing equations for additional scalars will be solved afterward and sequentially (that is, segregated from one another and from the coupled set) using the procedure described in General Scalar Transport Equation: Discretization and Solution (p. 960). Because the governing equations are nonlinear (and coupled), several iterations of the solution loop must be performed before a converged solution is obtained. Each iteration consists of the steps illustrated in Figure 24.2: Overview of the Density-Based Solution Method (p. 959) and outlined below:

1. Update the fluid properties based on the current solution. (If the calculation has just begun, the fluid properties will be updated based on the initialized solution.)

2. Solve the continuity, momentum, and (where appropriate) energy and species equations simultaneously.

3. Where appropriate, solve equations for scalars such as turbulence and radiation using the previously updated values of the other variables.

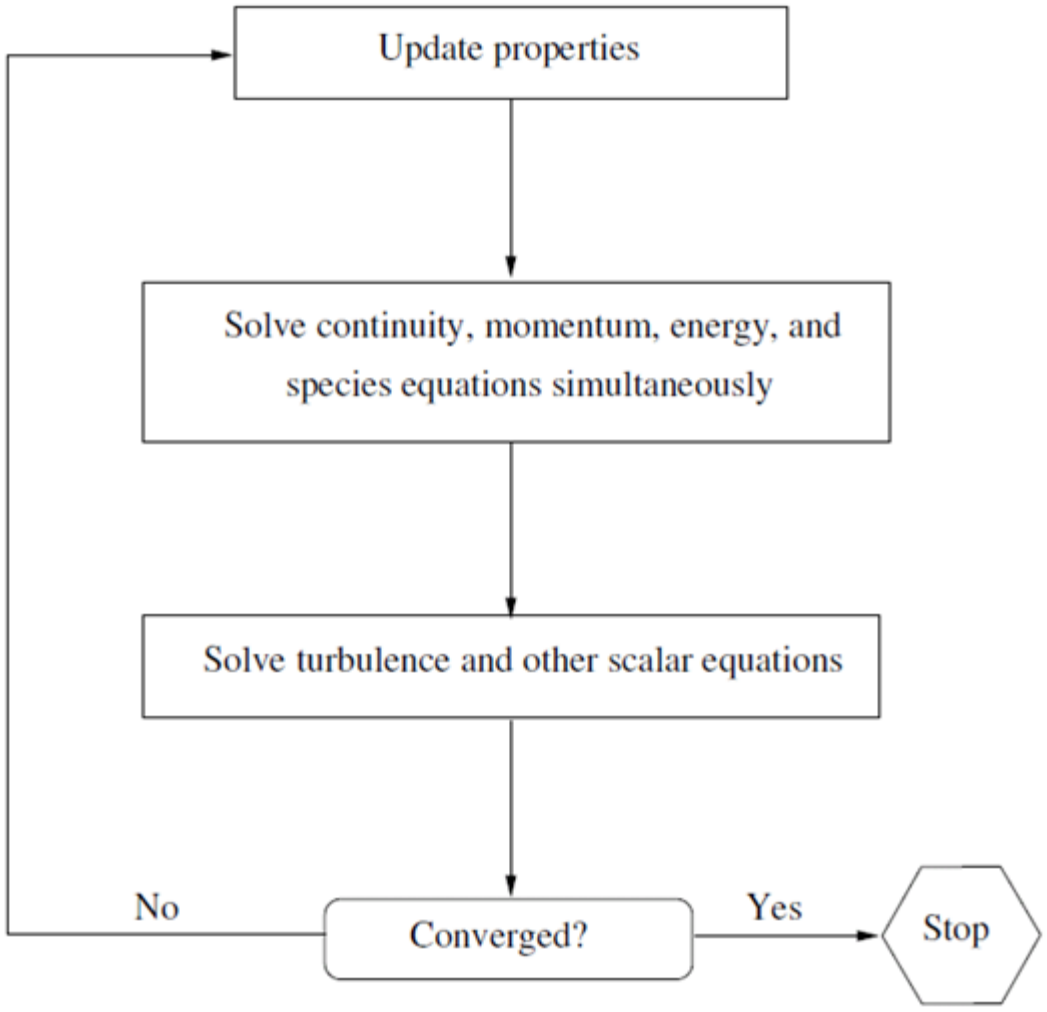
4. When interphase coupling is to be included, update the source terms in the appropriate continuous phase equations with a discrete phase trajectory calculation.

5. Check for convergence of the equation set.

These steps are continued until the convergence criteria are met.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Figure 24.2: Overview of the Density-Based Solution Method



In the density-based solution method, you can solve the coupled system of equations (continuity, momentum, energy and species equations if available) using, either the coupled-explicit formulation or the coupled-implicit formulation. The main distinction between the density-based explicit and implicit formulations is described next.

In the density-based solution methods, the discrete, nonlinear governing equations are linearized to produce a system of equations for the dependent variables in every computational cell. The resultant linear system is then solved to yield an updated flow-field solution.

The manner in which the governing equations are linearized may take an "implicit" or "explicit" form with respect to the dependent variable (or set of variables) of interest. By implicit or explicit we mean the following:

- implicit: For a given variable, the unknown value in each cell is computed using a relation that includes both existing and unknown values from neighboring cells. Therefore each unknown will appear in more than one equation in the system, and these equations must be solved simultaneously to give the unknown quantities.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

- explicit: For a given variable, the unknown value in each cell is computed using a relation that includes only existing values. Therefore each unknown will appear in only one equation in the system and the equations for the unknown value in each cell can be solved one at a time to give the unknown quantities.

In the density-based solution method you have a choice of using either an implicit or explicit linearization of the governing equations. This choice applies only to the coupled set of governing equations. Transport equations for additional scalars are solved segregated from the coupled set (such as turbulence, radiation, and so on). The transport equations are linearized and solved implicitly using the method described in General Scalar Transport Equation: Discretization and Solution (p. 960). Regardless of whether you choose the implicit or explicit methods, the solution procedure shown in Figure 24.2: Overview of the Density-Based Solution Method (p. 959) is followed.

If you choose the implicit option of the density-based solver, each equation in the coupled set of governing equations is linearized implicitly with respect to all dependent variables in the set. This will result in a system of linear equations with N equations for each cell in the domain, where N is the number of coupled equations in the set. Because there are N equations per cell, this is sometimes called a "block" system of equations.

A point implicit linear equation solver (Incomplete Lower Upper (ILU) factorization scheme or a symmetric block Gauss-Seidel) is used in conjunction with an algebraic multigrid (AMG) method to solve the resultant block system of equations for all N dependent variables in each cell. For example, linearization of the coupled continuity, x-, y-, z-momentum, and energy equation set will produce a system of equations in which p, u, v, w, and T are the unknowns. Simultaneous solution of this equation system (using the block AMG solver) yields at once updated pressure, u-, v-, w-velocity, and temperature fields.

In summary, the coupled implicit approach solves for all variables (p, u, v, w, T) in all cells at the same time.

If you choose the explicit option of the density-based solver, each equation in the coupled set of governing equations is linearized explicitly. As in the implicit option, this too will result in a system of equations with N equations for each cell in the domain and likewise, all dependent variables in the set will be updated at once. However, this system of equations is explicit in the unknown dependent variables. For example, the x-momentum equation is written such that the updated x velocity is a function of existing values of the field variables. Because of this, a linear equation solver is not needed. Instead, the solution is updated using a multi-stage (Runge-Kutta) solver. Here you have the additional option of employing a full approximation storage (FAS) multigrid scheme to accelerate the multi-stage solver.

In summary, the density-based explicit approach solves for all variables (p, u, v, w, T) one cell at a time.

Note that the FAS multigrid is an optional component of the explicit approach, while the AMG is a required element in both the pressure-based and density-based implicit approaches.

24.2. General Scalar Transport Equation: Discretization and Solution

Ansys Fluent uses a control-volume-based technique to convert a general scalar transport equation to an algebraic equation that can be solved numerically. This control volume technique consists of integ-

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

rating the transport equation about each control volume, yielding a discrete equation that expresses the conservation law on a control-volume basis.

Discretization of the governing equations can be illustrated most easily by considering the unsteady conservation equation for transport of a scalar quantity φ. This is demonstrated by the following equation written in integral form for an arbitrary control volume V as follows:

$$\int_V \frac{\partial \rho \varphi}{\partial t} dV + \oint \rho \varphi \vec{v} \cdot d\vec{A} = \oint \Gamma_\varphi \nabla \varphi \cdot d\vec{A} + \int_V S_\varphi dV \tag{24.1}$$

where

ρ = density

$\vec{v}$ = velocity vector (= $u\hat{i}+v\hat{j}$ in 2D)

$\vec{A}$ = surface area vector

Γ_φ = diffusion coefficient for φ

$\nabla\varphi$ = gradient of φ (= $(\partial\varphi/\partial x)\hat{i}+(\partial\varphi/\partial y)\hat{j}$ in 2D)

S_φ = source of φ per unit volume

Equation 24.1 (p. 961) is applied to each control volume, or cell, in the computational domain. The two-dimensional, triangular cell shown in Figure 24.3: Control Volume Used to Illustrate Discretization of a Scalar Transport Equation (p. 962) is an example of such a control volume. Discretization of Equation 24.1 (p. 961) on a given cell yields

$$\frac{\partial \rho \varphi}{\partial t} V + \sum_f^{N_{faces}} \rho_f \vec{v}_f \varphi_f \cdot \vec{A}_f = \sum_f^{N_{faces}} \Gamma_\varphi \nabla \varphi_f \cdot \vec{A}_f + S_\varphi V \tag{24.2}$$

where

N_{faces} = number of faces enclosing cell

φ_f = value of φ convected through face f

$\rho_f \vec{v}_f \cdot \vec{A}_f$ = mass flux through the face

$\vec{A}_f$ = area vector of face f, $|A|$ (= $|A_x\hat{i}+A_y\hat{j}|$ in 2D)

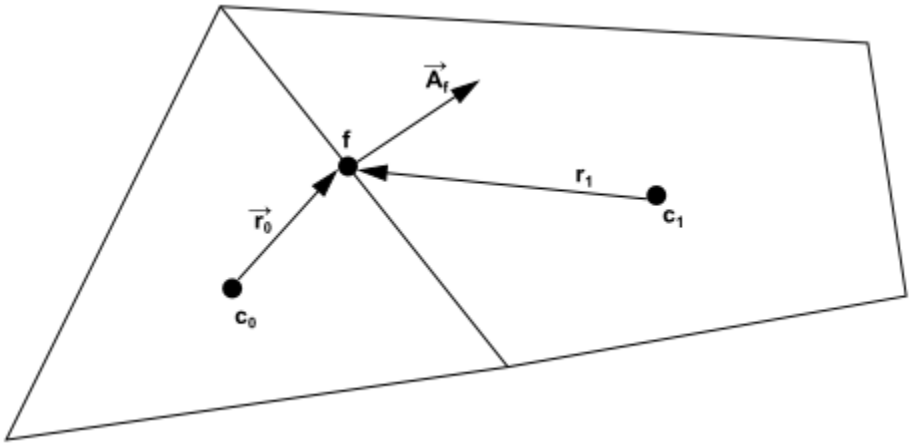
$\nabla\varphi_f$ = gradient of φ at face f

V = cell volume

Where $\frac{\partial \rho \varphi}{\partial t} V$ is defined in Temporal Discretization (p. 968). The equations solved by Ansys Fluent take the same general form as the one given above and apply readily to multi-dimensional, unstructured meshes composed of arbitrary polyhedra.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Figure 24.3: Control Volume Used to Illustrate Discretization of a Scalar Transport Equation



For more information, see the following section:

24.2.1. Solving the Linear System

24.2.1. Solving the Linear System

The discretized scalar transport equation (Equation 24.2 (p. 961)) contains the unknown scalar variable φ at the cell center as well as the unknown values in surrounding neighbor cells. This equation will, in general, be nonlinear with respect to these variables. A linearized form of Equation 24.2 (p. 961) can be written as

$$a_P \varphi = \sum_{nb} a_{nb} \varphi_{nb} + b \tag{24.3}$$

where the subscript nb refers to neighbor cells, and a_P and a_{nb} are the linearized coefficients for φ and φ_{nb}.

The number of neighbors for each cell depends on the mesh topology, but will typically equal the number of faces enclosing the cell (boundary cells being the exception).

Similar equations can be written for each cell in the mesh. This results in a set of algebraic equations with a sparse coefficient matrix. For scalar equations, Ansys Fluent solves this linear system using a point implicit (Gauss-Seidel) linear equation solver in conjunction with an algebraic multigrid (AMG) method that is described in Algebraic Multigrid (AMG) (p. 1009).

24.3. Discretization

Information is organized into the following subsections:

24.3.1. Spatial Discretization

24.3.2. Temporal Discretization

24.3.3. Evaluation of Gradients and Derivatives

24.3.4. Gradient Limiters

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

24.3.1. Spatial Discretization

By default, Ansys Fluent stores discrete values of the scalar φ at the cell centers (c_0 and c_1 in Figure 24.3: Control Volume Used to Illustrate Discretization of a Scalar Transport Equation (p. 962)). However, face values φ_f are required for the convection terms in Equation 24.2 (p. 961) and must be interpolated from the cell center values. This is accomplished using an upwind scheme.

Upwinding means that the face value φ_f is derived from quantities in the cell upstream, or "upwind," relative to the direction of the normal velocity v_n in Equation 24.2 (p. 961). Ansys Fluent allows you to choose from several upwind schemes: first-order upwind, second-order upwind, and QUICK. These schemes are described in First-Order Upwind Scheme (p. 963) – QUICK Scheme (p. 966).

The diffusion terms in Equation 24.2 (p. 961) are central-differenced and are always second-order accurate.

For information on how to use the various spatial discretization schemes, see Choosing the Spatial Discretization Scheme in the User's Guide.

24.3.1.1. First-Order Upwind Scheme

When first-order accuracy is desired, quantities at cell faces are determined by assuming that the cell-center values of any field variable represent a cell-average value and hold throughout the entire cell; the face quantities are identical to the cell quantities. Thus when first-order upwinding is selected, the face value φ_f is set equal to the cell-center value of φ in the upstream cell.

Important:

First-order upwind is available in the pressure-based and density-based solvers.

24.3.1.2. Second-Order Upwind Scheme

When second-order accuracy is desired, quantities at cell faces are computed using a multidimensional linear reconstruction approach [44] (p. 1055). In this approach, higher-order accuracy is achieved at cell faces through a Taylor series expansion of the cell-centered solution about the cell centroid. Thus when second-order upwinding is selected, the face value φ_f is computed using the following expression:

$$\varphi_{f,SOU} = \varphi + \nabla\varphi \cdot \vec{r} \tag{24.4}$$

where φ and $\nabla\varphi$ are the cell-centered value and its gradient in the upstream cell, and $\vec{r}$ is the displacement vector from the upstream cell centroid to the face centroid. This formulation requires the determination of the gradient $\nabla\varphi$ in each cell, as discussed in Evaluation of Gradients and Derivatives (p. 972). Finally, the gradient $\nabla\varphi$ is limited so that no new maxima or minima are introduced.

Important:

Second-order upwind is available in the pressure-based and density-based solvers.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

24.3.1.3. First- to Higher-Order Blending

In some instances, and at certain flow conditions, a converged solution to steady-state may not be possible with the use of higher-order discretization schemes due to local flow fluctuations (physical or numerical). On the other hand, a converged solution for the same flow conditions maybe possible with a first-order discretization scheme. For this type of flow and situation, if a better than first-order accurate solution is desired, then first- to higher-order blending can be used to obtain a converged steady-state solution.

The first-order to higher-order blending is applicable only when higher-order discretization is used. It is applicable with the following discretization schemes: second-order upwinding, central-differencing schemes, QUICK, and third-order MUSCL. The blending is not applicable to first-order, modified HRIC schemes, or the Geo-reconstruct and CICSAM schemes.

In the density-based solver, the blending is applied as a scaling factor to the reconstruction gradients. While in the pressure-based solver, the blending is applied to the higher-order terms for the convective transport variable.

To learn how to apply this option, refer to First- to Higher-Order Blending in the User's Guide.

24.3.1.4. Central-Differencing Scheme

A second-order-accurate central-differencing discretization scheme is available for the momentum equations when you are using a Scale-Resolving Simulation (SRS) turbulence model, such as LES. This scheme provides improved accuracy for SRS calculations.

The central-differencing scheme calculates the face value for a variable (φ_f) as follows:

$$\varphi_{f,CD}=\frac{1}{2}\left(\varphi_0+\varphi_1\right)+\frac{1}{2}\left(\nabla\varphi_0\cdot\vec{r}_0+\nabla\varphi_1\cdot\vec{r}_1\right) \tag{24.5}$$

where the indices 0 and 1 refer to the cells that share face f, $\nabla\varphi_{r,0}$ and $\nabla\varphi_{r,1}$ are the reconstructed gradients at cells 0 and 1, respectively, and $\vec{r}$ is the vector directed from the cell centroid toward the face centroid.

It is well known that central-differencing schemes can produce unbounded solutions and non-physical wiggles, which can lead to stability problems for the numerical procedure. These stability problems can often be avoided if a deferred correction is used for the central-differencing scheme. In this approach, the face value is calculated as follows:

$$\varphi_f=\underbrace{\varphi_{f,UP}}_{\text{implicit part}}+\underbrace{\left(\varphi_{f,CD}-\varphi_{f,UP}\right)}_{\text{explicit part}} \tag{24.6}$$

where UP stands for upwind. As indicated, the upwind part is treated implicitly while the difference between the central-difference and upwind values is treated explicitly. Provided that the numerical solution converges, this approach leads to pure second-order differencing.

Important:

The central differencing scheme is available only in the pressure-based solver.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

24.3.1.5. Bounded Central Differencing Scheme

The central differencing scheme described in Central-Differencing Scheme (p. 964) is an ideal choice for Scale-Resolving Simulation (SRS) turbulence models (such as LES) in view of its low numerical diffusion. However, it often leads to unphysical oscillations in the solution fields. In LES, the situation is exacerbated by usually very low subgrid-scale turbulent diffusivity. The bounded central differencing (BCD) scheme is essentially based on the normalized variable diagram (NVD) approach [353] (p. 1073) together with the convection boundedness criterion (CBC). The bounded central differencing scheme is a composite NVD-scheme that consists of a pure central differencing, a blended scheme of a central differencing and an upwind scheme, and the first-order upwind scheme. In the original NVD method [353] (p. 1073), the first-order upwind scheme is used whenever the CBC is violated, to suppress any local non-monotonicity. Experience with the application of this method to the scale-resolving simulation of turbulent flows has shown that it sometimes generates an overly high amount of numerical dissipation. Therefore, a tunable version of the BCD scheme is implemented in the pressure-based solver of Ansys Fluent. The boundedness strength of BCD can be controlled using a parameter, which allows to relax the strict CBC and to keep using the central differencing with the locally non-monotonous solution field, when the non-monotonicity is relatively low.

This parameter can be specified as a constant or an expression (see Solution Methods Task Page in the *Fluent User's Guide*) and can be changed within the range from 0 to 1. The maximum value of 1 corresponds to the strict CBC, which makes the BCD scheme more stable although more dissipative. The minimum value of 0 deactivates the BCD boundedness completely and turns the scheme to the pure central differencing. For simple flows (straight channel flow, free mixing layer flow, flat plate boundary layer flow), the parameter value of 0.25 has been found to be optimal and is selected as a default value. For complex flows, a higher value may be needed to provide an optimum balance between the stability and the resolution quality. Levels of around 0.75 have been found appropriate. Note that the tunable BCD scheme is available as a released feature only in the pressure-based solver.

The non-linearity of the BCD scheme may sometimes deteriorate the iteration convergence for the iterative time-advancement scheme. Freezing the weighting coefficients of the central differencing and upwind components of the BCD scheme helps to improve convergence in such situations. This freezing can be activated using the text command:

```
solve/set/advanced/bcd-weights-freeze
```

This command starts a dialog, asking for an iteration number, after which the BCD scheme weights are frozen at each time step.

Important:

The previous description is for the bounded central differencing scheme in the pressure-based solver; a similar—though not identical—BCD scheme is also available in the density-based solver.

Important:

In the pressure-based solver, the bounded central differencing scheme is the default momentum discretization scheme for the LES, DES, SAS, SDES, and SBES turbulence models.

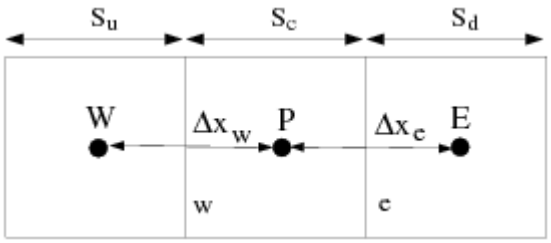
Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

24.3.1.6. QUICK Scheme

For quadrilateral and hexahedral meshes, where unique upstream and downstream faces and cells can be identified, Ansys Fluent also provides the QUICK scheme for computing a higher-order value of the convected variable φ at a face. QUICK-type schemes [354] (p. 1073) are based on a weighted average of second-order-upwind and central interpolations of the variable. For the face e in Figure 24.4: One-Dimensional Control Volume (p. 966), if the flow is from left to right, such a value can be written as

$$\varphi_e=\theta\left[\frac{S_d}{S_c+S_d}\varphi_P+\frac{S_c}{S_c+S_d}\varphi_E\right]+\left(1-\theta\right)\left[\frac{S_u+2S_c}{S_u+S_c}\varphi_P-\frac{S_c}{S_u+S_c}\varphi_W\right] \tag{24.7}$$

Figure 24.4: One-Dimensional Control Volume



$\theta=1$ in the above equation results in a central second-order interpolation while $\theta=0$ yields a second-order upwind value. The traditional QUICK scheme is obtained by setting $\theta=1/8$. The implementation in Ansys Fluent uses a variable, solution-dependent value of θ, chosen so as to avoid introducing new solution extrema.

The QUICK scheme will typically be more accurate on structured meshes aligned with the flow direction. Note that Ansys Fluent allows the use of the QUICK scheme for unstructured or hybrid meshes as well; in such cases the usual second-order upwind discretization scheme (described in Second-Order Upwind Scheme (p. 963)) will be used at the faces of non-hexahedral (or non-quadrilateral, in 2D) cells. The second-order upwind scheme will also be used at partition boundaries when the parallel solver is used.

Important:

The QUICK scheme is available in the pressure-based solver and when solving additional scalar equations in the density-based solver.

24.3.1.7. Third-Order MUSCL Scheme

This third-order convection scheme was conceived from the original MUSCL (Monotone Upstream-Centered Schemes for Conservation Laws) [662] (p. 1091) by blending a central differencing scheme and second-order upwind scheme as

$$\varphi_f=\theta\varphi_{f,CD}+\left(1-\theta\right)\varphi_{f,SOU} \tag{24.8}$$

where $\varphi_{f,CD}$ is defined in Equation 24.5 (p. 964), $\varphi_{f,SOU}$ is computed using the second-order upwind scheme as described in Second-Order Upwind Scheme (p. 963), and the blending factor θ is 1/3.

Unlike the QUICK scheme, which is applicable to structured hex meshes only, the MUSCL scheme is applicable to arbitrary meshes. Compared to the second-order upwind scheme, the third-order

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

MUSCL has a potential to improve spatial accuracy for all types of meshes by reducing numerical diffusion, most significantly for complex three-dimensional flows, and it is available for all transport equations.

Important:

The third-order MUSCL currently implemented in Ansys Fluent does not contain any Gradient limiter. As a result, it can produce undershoots and overshoots when the flow-field under consideration has discontinuities such as shock waves.

Important:

The MUSCL scheme is available in the pressure-based and density-based solvers.

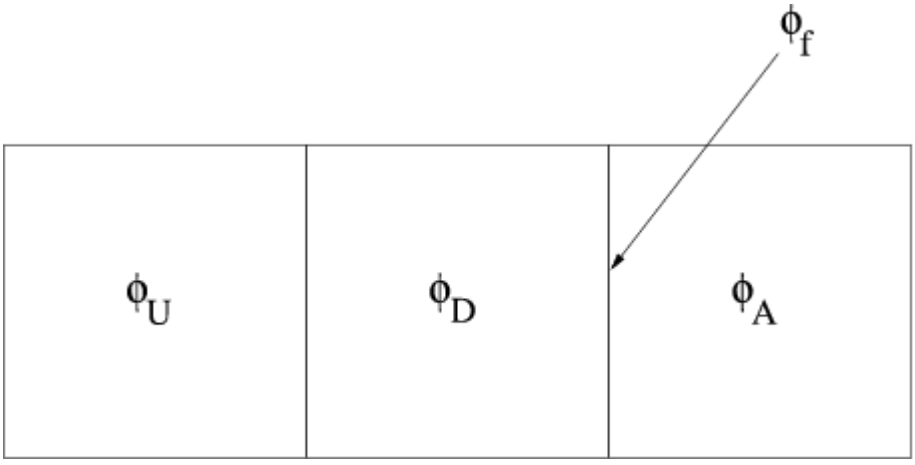
24.3.1.8. Modified HRIC Scheme

For simulations using the VOF multiphase model, upwind schemes are generally unsuitable for interface tracking because of their overly diffusive nature. Central differencing schemes, while generally able to retain the sharpness of the interface, are unbounded and often give unphysical results. In order to overcome these deficiencies, Ansys Fluent uses a modified version of the High Resolution Interface Capturing (HRIC) scheme. The modified HRIC scheme is a composite NVD scheme that consists of a nonlinear blend of upwind and downwind differencing [464] (p. 1080).

First, the normalized cell value of volume fraction, $\tilde{\varphi}_c$, is computed and is used to find the normalized face value, $\tilde{\varphi}_f$, as follows:

$$\tilde{\varphi}_c = \frac{\varphi_D - \varphi_U}{\varphi_A - \varphi_U} \tag{24.9}$$

Figure 24.5: Cell Representation for Modified HRIC Scheme



where A is the acceptor cell, D is the donor cell, and U is the upwind cell, and

$$\tilde{\varphi}_f = \begin{cases} \tilde{\varphi}_c & \tilde{\varphi}_c < 0 \text{ or } \tilde{\varphi}_c > 1 \\ 2\tilde{\varphi}_c & 0 \le \tilde{\varphi}_c \le 0.5 \\ 1 & 0.5 \le \tilde{\varphi}_c \le 1 \end{cases} \tag{24.10}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Here, if the upwind cell is not available (for example, unstructured mesh), an extrapolated value is used for φ_U. Directly using this value of $\tilde{\varphi}_f$ causes wrinkles in the interface, if the flow is parallel to the interface. So, Ansys Fluent switches to the ULTIMATE QUICKEST scheme (the one-dimensional bounded version of the QUICK scheme [353] (p. 1073)) based on the angle between the face normal and interface normal:

$$\tilde{\varphi}_f^{UQ} = \begin{cases} \tilde{\varphi}_c & \tilde{\varphi}_c < 0 \text{ or } \tilde{\varphi}_c > 1 \\ MIN\left(\tilde{\varphi}_f, \frac{6\tilde{\varphi}_c+3}{8}\right) & 0 \le \tilde{\varphi}_c \le 1 \end{cases} \tag{24.11}$$

This leads to a corrected version of the face volume fraction, $\tilde{\varphi}_f^*$:

$$\tilde{\varphi}_f^* = \tilde{\varphi}_f\sqrt{\cos\theta} + \left(1-\sqrt{\cos\theta}\right)\tilde{\varphi}_f^{UQ} \tag{24.12}$$

where

$$\cos\theta = \frac{\nabla\varphi\cdot\vec{d}}{|\nabla\varphi||\vec{d}|} \tag{24.13}$$

and $\vec{d}$ is a vector connecting cell centers adjacent to the face f.

The face volume fraction is now obtained from the normalized value computed above as follows:

$$\varphi_f = \tilde{\varphi}_f^*\left(\varphi_A - \varphi_U\right) + \varphi_U \tag{24.14}$$

The modified HRIC scheme provides improved accuracy for VOF calculations when compared to QUICK and second-order schemes, and is less computationally expensive than the **Geo-Reconstruct** scheme.

24.3.1.9. High Order Term Relaxation

Higher order schemes can be written as a first-order scheme plus additional terms for the higher-order scheme. The higher-order relaxation can be applied to these additional terms.

The under-relaxation of high order terms follows the standard formulation for any generic property φ

$$\varphi_{new} = \varphi_{old} + f\left(\varphi_{intermediate} - \varphi_{old}\right) \tag{24.15}$$

Where f is the under-relaxation factor. Note that the default value of f for steady-state cases is 0.25 and for transient cases is 0.75. The same factor is applied to all equations solved.

For information about this solver option, see High Order Term Relaxation (HOTR) in the User's Guide.

24.3.2. Temporal Discretization

For transient simulations, the governing equations must be discretized in both space and time. The spatial discretization for the time-dependent equations is identical to the steady-state case. Temporal discretization involves the integration of every term in the differential equations over a time step Δt. The integration of the transient terms is straightforward, as shown below.

A generic expression for the time evolution of a variable φ is given by

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$\frac{\partial \varphi}{\partial t} = F(\varphi) \tag{24.16}$$

where the function F incorporates any spatial discretization. If the time derivative is discretized using backward differences, the first-order accurate temporal discretization is given by

$$\frac{\varphi^{n+1} - \varphi^{n}}{\Delta t} = F(\varphi) \tag{24.17}$$

and the second-order discretization is given as described in Second-Order Time Integration Using a Variable Time Step Size (p. 970).

where

φ = a scalar quantity

$n+1$ = value at the next time level, $t+\Delta t$

n = value at the current time level, t

Once the time derivative has been discretized, a choice remains for evaluating $F(\varphi)$: in particular, which time level values of φ should be used in evaluating F.

24.3.2.1. Implicit Time Integration

One method is to evaluate $F(\varphi)$ at the future time level:

$$\frac{\varphi^{n+1} - \varphi^{n}}{\Delta t} = F(\varphi^{n+1}) \tag{24.18}$$

This is referred to as "implicit" integration since φ^{n+1} in a given cell is related to φ^{n+1} in neighboring cells through $F(\varphi^{n+1})$:

$$\varphi^{n+1} = \varphi^{n} + \Delta t F(\varphi^{n+1}) \tag{24.19}$$

This implicit equation can be solved iteratively at each time level before moving to the next time step.

The advantage of the fully implicit scheme is that it is unconditionally stable with respect to time step size.

24.3.2.2. Bounded Second-Order Implicit Time Integration

Any independent variable could be discretized in time as

$$\frac{\partial \varphi}{\partial t} = \frac{\varphi_{n+1/2} - \varphi_{n-1/2}}{dt} \tag{24.20}$$

$$\varphi_{n+1/2} = \varphi_{n} + \frac{1}{2}\beta_{n+1/2}(\varphi_{n} - \varphi_{n-1}) \tag{24.21}$$

$$\varphi_{n-1/2} = \varphi_{n-1} + \frac{1}{2}\beta_{n-1/2}(\varphi_{n-1} - \varphi_{n-2}) \tag{24.22}$$

Where n, $n-1$, $n-2$, $n+1/2$, $n-1/2$ are different time levels.

$\beta_{n+1/2}$ and $\beta_{n-1/2}$ are bounding factors for each variable at the $n+1/2$ and $n-1/2$ level.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Bounded variables include the following:

- Multiphase flows: Volume fraction
- Turbulent flows: Turbulence kinetic energy, dissipation rate, specific dissipation rate
- Reacting flows: Species mass fraction, premixed/non-premixed variables

24.3.2.2.1. Limitations

The following limitations exist when using the bounded second-order implicit formulation:

- It is not available with the density-based solver, only the pressure-based solver.
- It is not available with moving deforming meshes.
- It is only applicable to the implicit volume fraction discretization schemes for multiphase, and not to the explicit volume fraction schemes.
- It does not support the Singhal et al. expert cavitation model.

24.3.2.3. Second-Order Time Integration Using a Variable Time Step Size

Ansys Fluent's default second-order time integration schemes are based on a variable time step size formulation. This formulation introduces a generalized time derivative discretization that supports the use of variable time step sizes at any arbitrary time instance.

The generalized second-order discretization is given by

$$\frac{\left(1+\tau^{n+\frac{1}{2}}\right)\varphi^{n+1}-\left(1+\tau^{n+\frac{1}{2}}+\tau^{n-\frac{1}{2}}\right)\varphi^{n}+\left(\tau^{n-\frac{1}{2}}\right)\varphi^{n-1}}{2\Delta\tau^{n+1}}=F\left(\varphi\right) \tag{24.23}$$

In the previous equation, τ is a time step size ratio defined as

$$\tau^{n+\frac{1}{2}}=\frac{\Delta t^{n+1}}{\left(\Delta t^{n+1}+\Delta t^{n}\right)}$$

and

$$\tau^{n-\frac{1}{2}}=\frac{\Delta t^{n}}{\left(\Delta t^{n}+\Delta t^{n-1}\right)}$$

where

Δt^{n+1} = current time step size

Δt^{n} = previous time step size

Δt^{n-1} = time step size two time levels prior to the current

When using fixed time steps,

$$\Delta t^{n+1}=\Delta t^{n}=\Delta t^{n-1}\equiv\Delta t$$

and thus

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$\tau^{n+\frac{1}{2}}=\tau^{n-\frac{1}{2}}=\frac{1}{2}$$

Consequently, the generalized equation becomes a fixed time step size formulation defined as:

$$\frac{3\varphi^{n+1}-4\varphi^{n}+\varphi^{n-1}}{2\Delta t}=F(\varphi) \tag{24.24}$$

For the bounded second-order implicit formulation, the time step size ratio must be included as a factor acting on $\beta_{n+\frac{1}{2}}$ and $\beta_{n-\frac{1}{2}}$, the bounding factors for each variable at $n+\frac{1}{2}$ and $n-\frac{1}{2}$ time levels, and so Equation 24.21 (p. 969) and Equation 24.22 (p. 969) become:

$$\varphi_{n+1/2}=\varphi_{n}+(\tau\beta)_{n+\frac{1}{2}}(\varphi_{n}-\varphi_{n-1}) \tag{24.25}$$

$$\varphi_{n-1/2}=\varphi_{n-1}+(\tau\beta)_{n-\frac{1}{2}}(\varphi_{n-1}-\varphi_{n-2}) \tag{24.26}$$

For the density-based solver, the variables ε_0, ε_1, and ε_2 in Equation 24.98 (p. 999) are redefined as the following, in order to extend the time derivative to the variable time step size formulation.

$$\left\{\begin{array}{c}\varepsilon_0=\left(1+\tau^{n+\frac{1}{2}}\right),\\ \varepsilon_1=\left(1+\tau^{n-\frac{1}{2}}+\tau^{n+\frac{1}{2}}\right),\\ \varepsilon_2=\left(\tau^{n-\frac{1}{2}}\right)\end{array}\right\} \tag{24.27}$$

Important:

The variable time step size formulation for second-order time integration is only available for fixed or sliding meshes. For dynamic meshes, the fixed time step size formulation (Equation 24.24 (p. 971)) is used: for such cases, do not run a series of simulations in which you vary the time step size, as doing so creates an error that reduces with a reduction of the time step jump.

24.3.2.4. Explicit Time Integration

The explicit time integration is only available when using the explicit density-based solver, in which $F(\varphi)$ is evaluated explicitly (based on the existing solution of the dependent variable ϕ:

$$\frac{\varphi^{n+1}-\varphi^{n}}{\Delta t}=F(\varphi^{n}) \tag{24.28}$$

and is referred to as "explicit" integration since φ^{n+1} can be expressed explicitly in terms of the existing solution values, φ^{n}:

$$\varphi^{n+1}=\varphi^{n}+\Delta tF(\varphi^{n}) \tag{24.29}$$

Here, the time step Δt is restricted to the stability limit of the underlying solver (that is, a time step is limited by the Courant-Friedrichs-Lewy condition). In order to be time-accurate, all cells in the domain must use the same time step. For stability, this time step must be the minimum of all the local time steps in the domain. This method is also referred to as "global time stepping".

The use of explicit time stepping is fairly restrictive. It is used primarily to capture the transient behavior of moving waves, such as shocks, because it is more accurate and less expensive than the

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

implicit time stepping methods in such cases. You *cannot* use explicit time stepping in the following cases:

- Calculations with the pressure-based solver or density-based implicit formulation: The explicit time stepping formulation is available only with the density-based explicit formulation. Ansys Fluent also uses multi-stage Runge-Kutta explicit time integration for the density-based solver, as detailed in Steady-State Flow Solution Methods (p. 995) and Unsteady Flows Solution Methods (p. 998).
- Incompressible flow: Explicit time stepping cannot be used to compute time-accurate incompressible flows.
- FAS multigrid and residual smoothing cannot be used with explicit time stepping because they destroy the time accuracy of the underlying solver.

24.3.3. Evaluation of Gradients and Derivatives

Gradients are needed not only for constructing values of a scalar at the cell faces, but also for computing secondary diffusion terms and velocity derivatives. The gradient $\nabla\varphi$ of a given variable φ is used to discretize the convection and diffusion terms in the flow conservation equations. The gradients are computed in Ansys Fluent according to the following methods:

- Green-Gauss Cell-Based
- Green-Gauss Node-Based
- Least Squares Cell-Based

To learn how to apply the various gradients, see Choosing the Spatial Discretization Scheme in the User's Guide.

24.3.3.1. Green-Gauss Theorem

When the Green-Gauss theorem is used to compute the gradient of the scalar φ at the cell center $c0$, the following discrete form is written as

$$\left(\nabla\varphi\right)_{c0} = \frac{1}{\vec{v}}\sum_{f}\overline{\varphi}_{f}\vec{A}_{f} \tag{24.30}$$

where φ_f is the value of φ at the cell face centroid, computed as shown in the sections below. The summation is over all the faces enclosing the cell.

24.3.3.2. Green-Gauss Cell-Based Gradient Evaluation

By default, the face value, $\overline{\varphi}_f$, in Equation 24.30 (p. 972) is taken from the arithmetic average of the values at the neighboring cell centers, that is,

$$\overline{\varphi}_{f}=\frac{\varphi_{c0}+\varphi_{c1}}{2} \tag{24.31}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

24.3.3.3. Green-Gauss Node-Based Gradient Evaluation

Alternatively, $\overline{\varphi}_f$ can be computed by the arithmetic average of the nodal values on the face.

$$\overline{\varphi}_f = \frac{1}{N_f} \sum_{n}^{N_f} \overline{\varphi}_n \tag{24.32}$$

where N_f is the number of nodes on the face.

The nodal values, $\overline{\varphi}_n$ in Equation 24.32 (p. 973), are constructed from the weighted average of the cell values surrounding the nodes, following the approach originally proposed by Holmes and Connel [254] (p. 1067) and Rauch et al. [542] (p. 1084). This scheme reconstructs exact values of a linear function at a node from surrounding cell-centered values on arbitrary unstructured meshes by solving a constrained minimization problem, preserving a second-order spatial accuracy.

The node-based gradient is known to be more accurate than the cell-based gradient particularly on irregular (skewed and distorted) unstructured meshes, however, it is relatively more expensive to compute than the cell-based gradient scheme.

In the density-based solver, the stability of the node-based gradient can be reduced by the presence of tetrahedral elements along domain boundaries. To regain robustness on tetrahedral and mixed meshes, it is recommended to select the extended node-based boundary option, available under the text command:

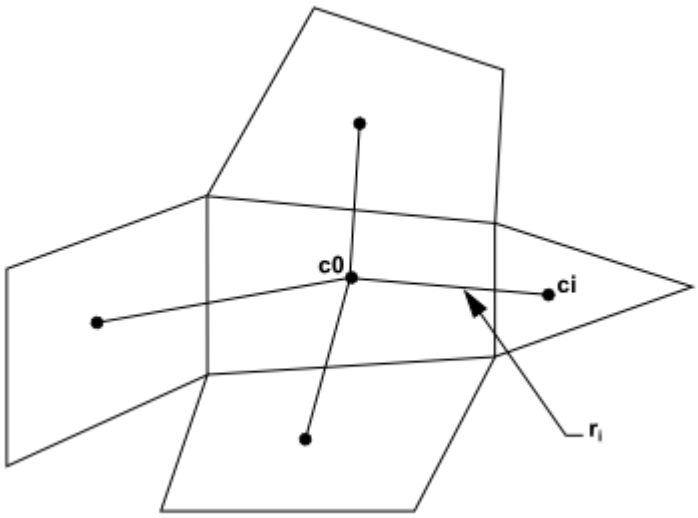
```
solve/set/nb-gradient-boundary-option? yes yes
```

24.3.3.4. Least Squares Cell-Based Gradient Evaluation

In this method the solution is assumed to vary linearly. In Figure 24.6: Cell Centroid Evaluation (p. 973), the change in cell values between cell $c0$ and ci along the vector δr_i from the centroid of cell $c0$ to cell ci, can be expressed as

$$\left(\nabla\varphi\right)_{c0} \cdot \Delta r_i = \left(\varphi_{ci} - \varphi_{c0}\right) \tag{24.33}$$

Figure 24.6: Cell Centroid Evaluation



If we write similar equations for each cell surrounding the cell c0, we obtain the following system written in compact form:

$$[J](\nabla\varphi)_{c0}=\Delta\varphi \tag{24.34}$$

Where [J] is the coefficient matrix that is purely a function of geometry.

The objective here is to determine the cell gradient ($\nabla\varphi_0=\varphi_x\hat{i}+\varphi_y\hat{j}+\varphi_z\hat{k}$) by solving the minimization problem for the system of the non-square coefficient matrix in a least-squares sense.

The above linear-system of equation is over-determined and can be solved by decomposing the coefficient matrix using the Gram-Schmidt process [19] (p. 1054). This decomposition yields a matrix of weights for each cell. Thus for our cell-centered scheme this means that the three components of the weights ($W^x_{i0}, W^y_{i0}, W^z_{i0}$) are produced for each of the faces of cell c0.

Therefore, the gradient at the cell center can then be computed by multiplying the weight factors by the difference vector $\Delta\varphi=\left(\varphi_{c1}-\varphi_{c0}\right)$,

$$\left(\varphi_x\right)_{c0} = \sum_{i=1}^{n} W^x_{i0}\cdot\left(\varphi_{ci}-\varphi_{c0}\right) \tag{24.35}$$

$$\left(\varphi_y\right)_{c0} = \sum_{i=1}^{n} W^y_{i0}\cdot\left(\varphi_{ci}-\varphi_{c0}\right) \tag{24.36}$$

$$\left(\varphi_z\right)_{c0} = \sum_{i=1}^{n} W^z_{i0}\cdot\left(\varphi_{ci}-\varphi_{c0}\right) \tag{24.37}$$

On irregular (skewed and distorted) unstructured meshes, the accuracy of the least-squares gradient method is comparable to that of the node-based gradient (and both are much more superior compared to the cell-based gradient). However, it is less expensive to compute the least-squares gradient than the node-based gradient. Therefore, it has been selected as the default gradient method in the Ansys Fluent solver.

24.3.4. Gradient Limiters

Gradient limiters, also known as slope limiters, are used on the second-order upwind (SOU) scheme to prevent spurious oscillations, which would otherwise appear in the solution flow field near shocks, discontinuities, or near rapid local changes in the flow field. The gradient limiter attempts to invoke and enforce the monotonicity principle by prohibiting the linearly reconstructed field variable on the cell faces to exceed the maximum or minimum values of the neighboring cells.

There are three gradient limiters in the Ansys Fluent solvers:

- Standard limiter
- Multidimensional limiter
- Differentiable limiter

Gradient limiters can be categorized into two general groups: non-differentiable limiters and differentiable limiters. Both, the standard limiter and multidimensional limiter are of the non-differentiable form, since they use minimum and maximum types of functions for limiting the solution variables.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

The third limiter in Ansys Fluent, as the name indicates, is a differentiable type of limiter, which uses a smooth function to impose the monotonicity principle.

For each of the above mentioned limiter methods, Ansys Fluent provides two limiting directions:

- cell to face limiting, where the limited value of the reconstruction gradient is determined at cell face centers. This is the default method.
- cell to cell limiting, where the limited value of the reconstruction gradient is determined along a scaled line between two adjacent cell centroids. On an orthogonal mesh (or when cell-to-cell direction is parallel to face area direction) this method becomes equivalent to the default cell to face method. For smooth field variation, cell to cell limiting may provide less numerical dissipation on meshes with skewed cells.
- cell to node limiting, where the limited value of the reconstruction gradient is determined at the vertices of a cell. This limiting direction is more conservative than the default cell to face strategy, leading to improved monotonicity and robustness when computing strong gradients on unstructured grids. It is particularly advantageous for meshes containing stretched, skewed, or high aspect-ratio cells.

For more information about how to access the limiter functions in Ansys Fluent through the GUI or TUI, see Selecting Gradient Limiters in the User's Guide.

Important:

On unstructured meshes, Ansys Fluent uses the scalar form of the gradient limiter given by the following equation:

$$\Phi_{f_{SOU}} = \Phi + \psi \nabla \varphi \cdot r \tag{24.38}$$

Where ψ is a scalar value that limits the gradient $\nabla \Phi$.

24.3.4.1. Standard Limiter

The standard limiter is the default limiter function in Ansys Fluent and is derived from the work of Barth and Jespersen [44] (p. 1055). This limiter is of a non-differentiable type and uses the Minmod function (Minimum Modulus) to limit and clip the reconstructed solution overshoots and undershoots on the cell faces.

24.3.4.2. Multidimensional Limiter

The multidimensional limiter in Ansys Fluent [303] (p. 1070) has a similar form to the standard limiter. Since the multidimensional limiter uses a Minmod function for limiting the gradient, it is also classified as a non-differentiable type of limiter. However, in the standard limiter formulation, if limiting took place on any face of the cell, then this will cause the cell gradient to be clipped in an equal manner, in all directions, regardless of whether or not limiting is needed on the other cell faces. This limiting method is rather severe and adds unnecessary dissipation to the numerical scheme. The multidimensional limiter, on the other hand, attempts to lessen the severity of the gradient limiting by carefully examining the gradient on each cell and clipping only the normal components of the gradient to the cell faces. For this procedure to work on a scalar form limiter, the normal components of gradients on cell faces are first sorted out in ascending order of their magnitude

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

so that only the necessary clipping can be applied. The multidimensional limiter is therefore less dissipative than the standard limiter.

24.3.4.3. Differentiable Limiter

One disadvantage with non-differentiable limiters is that they tend to stall the apparent residual's convergence after a few orders of reduction in residual magnitude. Note that this does not mean that the solution is not converging, but rather the solution continues to converge while the residuals are stalling. This annoying behavior can be directly traced to the non-differentiable nature of the limiting functions. Therefore, the differentiable limiter uses a smooth function to impose the monotonicity condition while allowing the residuals to converge. The differentiable limiter used in Ansys Fluent is a modified form [689] (p. 1093) of a limiter which was originally proposed by Venkatakrishnan [666] (p. 1091).

Important:

Ansys Fluent uses gradient or slope limiters and not flux limiters. Gradient limiters are applied to the gradients of the variable field being linearly reconstructed at the cell faces, while flux limiters are used on the system fluxes.

24.4. Pressure-Based Solver

In this section, special practices related to the discretization of the momentum and continuity equations and their solution by means of the pressure-based solver are addressed. Information is organized into the following subsections:

24.4.1. Discretization of the Momentum Equation

24.4.2. Discretization of the Continuity Equation

24.4.3. Pressure-Velocity Coupling

24.4.4. Steady-State Iterative Algorithm

24.4.5. Time-Advancement Algorithm

24.4.6. Correction Form Discretization of the Momentum Equations

These special practices are most easily described by considering the steady-state continuity and momentum equations in integral form:

$$\oint \rho \vec{v} \cdot d\vec{A} = 0 \tag{24.39}$$

$$\oint \rho \vec{v} \vec{v} \cdot d\vec{A} = -\oint pI \cdot d\vec{A} + \oint \bar{\bar{\tau}} \cdot d\vec{A} + \int_V \vec{F} dV \tag{24.40}$$

where I is the identity matrix, $\bar{\bar{\tau}}$ is the stress tensor, and $\vec{F}$ is the force vector.

24.4.1. Discretization of the Momentum Equation

The discretization scheme described in Discretization (p. 962) for a scalar transport equation is also used to discretize the momentum equations. For example, the x-momentum equation can be obtained by setting $\varphi = u$:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$a_P u = \sum_{nb} a_{nb} u_{nb} + \sum p_f A \cdot \hat{i} + S \tag{24.41}$$

If the pressure field and face mass fluxes are known, Equation 24.41 (p. 977) can be solved in the manner outlined in Discretization (p. 962), and a velocity field obtained. However, the pressure field and face mass fluxes are not known a priori and must be obtained as a part of the solution. There are important issues with respect to the storage of pressure and the discretization of the pressure gradient term; these are addressed next.

Ansys Fluent uses a co-located scheme, whereby pressure and velocity are both stored at cell centers. However, Equation 24.41 (p. 977) requires the value of the pressure at the face between cells $c0$ and $c1$, shown in Figure 24.3: Control Volume Used to Illustrate Discretization of a Scalar Transport Equation (p. 962). Therefore, an interpolation scheme is required to compute the face values of pressure from the cell values.

24.4.1.1. Pressure Interpolation Schemes

Ansys Fluent offers the following options for interpolating the pressure values at the faces. By default the **Second Order** scheme is used except in the case of mixture or VOF multiphase simulations in which case **PRESTO!** is the default.

- The **Linear** scheme computes the face pressure as the average of the pressure values in the adjacent cells.
- The **Standard** scheme interpolates the pressure values at the faces using momentum equation coefficients [551] (p. 1085):

$$P_f = \frac{\frac{P_{c_0}}{a_{p,c_0}} + \frac{P_{c_1}}{a_{p,c_1}}}{\frac{1}{a_{p,c_0}} + \frac{1}{a_{p,c_1}}} \tag{24.42}$$

  This procedure works well as long as the pressure variation between cell centers is smooth. When there are jumps or large gradients in the momentum source terms between control volumes, the pressure profile has a high gradient at the cell face, and cannot be interpolated using this scheme. If this scheme is used, the discrepancy shows up in overshoots/undershoots of cell velocity.

  Flows for which the standard pressure interpolation scheme will have trouble include flows with large body forces, such as in strongly swirling flows, in high-Rayleigh-number natural convection and the like. In such cases, it is necessary to pack the mesh in regions of high gradient to resolve the pressure variation adequately.

  Another source of error is that Ansys Fluent assumes that the normal pressure gradient at the wall is zero. This is valid for boundary layers, but not in the presence of body forces or curvature. Again, the failure to correctly account for the wall pressure gradient is manifested in velocity vectors pointing in/out of walls.

- The **Second Order** scheme reconstructs the face pressure using a central differencing scheme. The pressure values at the faces are given by:

$$P_f = \frac{1}{2}\left(P_{c_0} + P_{c_1}\right) + \frac{1}{2}\left(\nabla P_{c_0} \cdot \vec{r}_{c_0} + \nabla P_{c_1} \cdot \vec{r}_{c_1}\right) \tag{24.43}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

This scheme may provide improved accuracy over the **Standard** and **Linear** schemes.

For flows simulated using the Eulerian multiphase model, the **Second Order** scheme is not applicable in this form due to large changes in flow properties that may occur across a face. However, by incorporating the effect of the pressure gradient on face pressure into Equation 24.43 (p. 977) and combining it with the **Standard** scheme (Equation 24.42 (p. 977)), the **Second Order** scheme for Eulerian multiphase flows can be obtained as:

$$P_f = \frac{\frac{\left(P_{c_0} + \nabla P_{c_0} \vec{r}_{c_0}\right)}{ap_{c,0}} + \frac{P_{c_1} + \nabla P_{c_1} \vec{r}_{c_1}}{ap_{c,1}}}{\left(\frac{1}{ap_{c,0}} + \frac{1}{ap_{c,1}}\right)} \tag{24.44}$$

This scheme is more robust than **PRESTO!** for Eulerian multiphase cases where the mesh topology deviates largely from orthogonality.

- The **Body Force Weighted** scheme computes the face pressure by assuming that the normal gradient of the difference between pressure and body forces is constant. This works well if the body forces are known a priori in the momentum equations (for example, buoyancy and axisymmetric swirl calculations). For Eulerian Multi-Fluid VOF cases where body forces dominate and the mesh deviates largely from orthogonality, the **Body Force Weighted** scheme is more robust than **PRESTO!**.

  Important:

  When a case contains porous media, the body-force-weighted scheme is applied only for non-porous faces, where the scheme takes into account the discontinuity of explicit body forces (for example, gravity, swirl, Coriolis) and the discontinuity of pressure gradients for flows with rapidly changing densities (for example, natural convection, VOF). All interior and exterior porous faces are treated with a special scheme that preserves the continuity of the normal velocity across cell faces in spite of the discontinuity of the resistance.

- The **Modified Body Force Weighted** scheme is an extended variant of the **Body Force Weighted** scheme, which overcomes the shortcomings of the **Body Force Weighted** scheme for highly viscous and rotating flows and provides better solution stability and robustness in general. This scheme is available with the VOF and Mixture multiphase models only.

- The **PRESTO!** (PREssure STaggering Option) scheme uses the discrete continuity balance for a "staggered" control volume about the face to compute the "staggered" (that is, face) pressure. This procedure is similar in spirit to the staggered-grid schemes used with structured meshes [504] (p. 1082). Note that for triangular, tetrahedral, hybrid, and polyhedral meshes, comparable accuracy is obtained using a similar algorithm. The PRESTO! scheme is available for all meshes.

For recommendations on when to use these alternate schemes, see Choosing the Pressure Interpolation Scheme in the User's Guide.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

24.4.2. Discretization of the Continuity Equation

Equation 24.39 (p. 976) may be integrated over the control volume in Figure 24.3: Control Volume Used to Illustrate Discretization of a Scalar Transport Equation (p. 962) to yield the following discrete equation

$$\sum_{f}^{N_{faces}} J_f A_f = 0 \tag{24.45}$$

where J_f is the mass flux through face f, ρv_n.

In order to proceed further, it is necessary to relate the face value of velocity, $\vec{v}_n$, to the velocity at the centers of the two adjacent cells. Linear interpolation of cell-centered velocities to the face results in unphysical checkerboarding of pressure. Ansys Fluent uses high-order interpolation methods to calculate face flux, $\hat{J}_f$, and a pressure gradient correction is applied following a similar method outlined by Rhie and Chow [551] (p. 1085) to prevent checkerboarding. Two interpolation methods are available for calculating the face value of velocity: distance-based linear interpolation and momentum-based averaging [415] (p. 1077). Using this procedure, the face flux, J_f, may be written as

$$J_f = \hat{J}_f + d_f\left(P_{c_0} - P_{c_1}\right) \tag{24.46}$$

where

$$d_f\left(P_{c_0} - P_{c_1}\right) = d_f\left(\left(p_{c_0} + \left(\nabla p\right)_{c_0} \cdot \vec{r}_0\right) - \left(p_{c_1} + \left(\nabla p\right)_{c_1} \cdot \vec{r}_1\right)\right) \tag{24.47}$$

The definition of $\hat{J}_f$ depends on the method:

$$\text{distance-based:} \quad \hat{J}_f = \frac{\rho_f}{2}\left(\vec{v}_{c_0} + \vec{v}_{c_1}\right) \cdot \hat{n} + J_{ho,\,d} \tag{24.48}$$

$$\text{momentum-based:} \quad \hat{J}_f = \rho_f\left(\frac{a_{p,c_0}\vec{v}_{c_0} + a_{p,c_1}\vec{v}_{c_1}}{a_{p,c_0} + a_{p,c_1}}\right) \cdot \hat{n} + J_{ho,\,a_p} \tag{24.49}$$

The term d_f is a function of $\bar{a}_P$, the average of the momentum equation a_P coefficients (from Equation 24.41 (p. 977)) for the cells on either side of face f. p_{c_0}, p_{c_1} and $\vec{v}_{c_0}$, $\vec{v}_{c_1}$ are the pressures and velocities, respectively, within the two cells on either side of the face. $J_{ho,\,d}$ is a high-order correction for non-uniform and/or non-orthogonal meshes based on distances between the cell centers and the face center. Similarly, $J_{ho,\,a_p}$ is a high-order correction for non-uniform and/or non-orthogonal meshes based on the momentum coefficients of cells adjacent to the face.

24.4.2.1. Density Interpolation Schemes

For incompressible flows, Ansys Fluent uses arithmetic averaging for density. For compressible flow calculations (that is, calculations that use the ideal gas law for density), Ansys Fluent applies upwind interpolation of density at cell faces. Several interpolation schemes are available for the density upwinding at cell faces: first-order upwind, second-order upwind (default), QUICK, MUSCL, and when applicable, central differencing and bounded central differencing.

The first-order upwind scheme (based on [289] (p. 1069)) sets the density at the cell face to be the upstream cell-center value. This scheme provides stability for the discretization of the pressure-

correction equation, and gives good results for most classes of flows. Although this scheme provides the best stability for compressible flow calculations, it gives very diffusive representations of shocks.

The second-order upwind scheme provides stability for supersonic flows and captures shocks better than the first-order upwind scheme. The QUICK scheme for density is similar to the QUICK scheme used for other variables. See QUICK Scheme (p. 966) for details.

Important:

In the case of multiphase flows, the selected density scheme is applied to the compressible phase and arithmetic averaging is used for incompressible phases.

Important:

If solution instability is encountered for compressible flow, it is recommended that you achieve a solution with a first order scheme and then switch to a higher order schemes.

For recommendations on choosing an appropriate density interpolation scheme for your compressible flow, see Choosing the Density Interpolation Scheme in the User's Guide.

24.4.3. Pressure-Velocity Coupling

Pressure-velocity coupling is achieved by using Equation 24.46 (p. 979) to derive an additional condition for pressure by reformatting the continuity equation (Equation 24.45 (p. 979)). The pressure-based solver allows you to solve your flow problem in either a segregated or coupled manner. Ansys Fluent provides the option to choose among five pressure-velocity coupling algorithms: SIMPLE, SIMPLEC, PISO, Coupled, and (for unsteady flows using the non-iterative time advancement scheme (NITA)) Fractional Step (FSM). All the aforementioned schemes, except the "coupled" scheme, are based on the predictor-corrector approach. For instructions on how to select these algorithms, see Choosing the Pressure-Velocity Coupling Method in the User's Guide.

Note that **SIMPLE**, **SIMPLEC**, **PISO**, and **Fractional Step** use the pressure-based segregated algorithm, while **Coupled** uses the pressure-based coupled solver.

Important:

The pressure-velocity coupling schemes that are applicable when using the Eulerian multiphase model are **Phase Coupled SIMPLE**, **Multiphase Coupled**, and **Full Multiphase Coupled**. These are discussed in detail in Selecting the Pressure-Velocity Coupling Method in the User's Guide.

24.4.3.1. Segregated Algorithms

24.4.3.1.1. SIMPLE

The SIMPLE algorithm uses a relationship between velocity and pressure corrections to enforce mass conservation and to obtain the pressure field.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

If the momentum equation is solved with a guessed pressure field p^*, the resulting face flux, J_f^*, computed from Equation 24.46 (p. 979)

$$J_f^* = \hat{J}_f^* + d_f\left(p_{c0}^* - p_{c1}^*\right) \tag{24.50}$$

does not satisfy the continuity equation. Consequently, a correction J_f' is added to the face flux J_f^* so that the corrected face flux, J_f

$$J_f = J_f^* + J_f' \tag{24.51}$$

satisfies the continuity equation. The SIMPLE algorithm postulates that J_f' be written as

$$J_f' = d_f\left(p_{c0}' - p_{c1}'\right) \tag{24.52}$$

where p' is the cell pressure correction.

The SIMPLE algorithm substitutes the flux correction equations (Equation 24.51 (p. 981) and Equation 24.52 (p. 981)) into the discrete continuity equation (Equation 24.45 (p. 979)) to obtain a discrete equation for the pressure correction p' in the cell:

$$a_P p' = \sum_{nb} a_{nb} p_{nb}' + b \tag{24.53}$$

where the source term b is the net flow rate into the cell:

$$b = \sum_{f}^{N_{faces}} J_f^* A_f \tag{24.54}$$

The pressure-correction equation (Equation 24.53 (p. 981)) may be solved using the algebraic multigrid (AMG) method described in Algebraic Multigrid (AMG) (p. 1009). Once a solution is obtained, the cell pressure and the face flux are corrected using

$$p = p^* + \alpha_p p' \tag{24.55}$$

$$J_f = J_f^* + d_f\left(p_{c0}' - p_{c1}'\right) \tag{24.56}$$

Here α_p is the under-relaxation factor for pressure (see Under-Relaxation of Variables (p. 985) for information about under-relaxation). The corrected face flux, J_f, satisfies the discrete continuity equation identically during each iteration.

24.4.3.1.2. SIMPLEC

A number of variants of the basic SIMPLE algorithm are available in the literature. In addition to SIMPLE, Ansys Fluent offers the SIMPLEC (SIMPLE-Consistent) algorithm [664] (p. 1091). SIMPLE is the default for transient simulations, but many problems will benefit from the use of SIMPLEC, as described in SIMPLE vs. SIMPLEC in the User's Guide.

The SIMPLEC procedure is similar to the SIMPLE procedure outlined above. The only difference lies in the expression used for the face flux correction, J_f'. As in SIMPLE, the correction equation may be written as

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$J_f = J_f^* + d_f\left(p'_{c0} - p'_{c1}\right) \tag{24.57}$$

However, the coefficient d_f is redefined as a function of $\left(\overline{a_P - \sum_{nb} a_{nb}}\right)$. The use of this modified correction equation has been shown to accelerate convergence in problems where pressure-velocity coupling is the main deterrent to obtaining a solution.

24.4.3.1.2.1. Skewness Correction

For meshes with some degree of skewness, the approximate relationship between the correction of mass flux at the cell face and the difference of the pressure corrections at the adjacent cells is very rough. Since the components of the pressure-correction gradient along the cell faces are not known in advance, an iterative process similar to the PISO neighbor correction described below is desirable. After the initial solution of the pressure-correction equation, the pressure-correction gradient is recalculated and used to update the mass flux corrections. This process, which is referred to as "skewness correction", significantly reduces convergence difficulties associated with highly distorted meshes. The SIMPLEC skewness correction allows Ansys Fluent to obtain a solution on a highly skewed mesh in approximately the same number of iterations as required for a more orthogonal mesh. Note that the skewness correction can use a default or an enhanced formulation.

24.4.3.1.3. PISO

The Pressure-Implicit with Splitting of Operators (PISO) pressure-velocity coupling scheme, part of the SIMPLE family of algorithms, is based on the higher degree of the approximate relation between the corrections for pressure and velocity. One of the limitations of the SIMPLE and SIMPLEC algorithms is that new velocities and corresponding fluxes do not satisfy the momentum balance after the pressure-correction equation is solved. As a result, the calculation must be repeated until the balance is satisfied. To improve the efficiency of this calculation, the PISO algorithm performs two additional corrections: neighbor correction and skewness correction.

24.4.3.1.3.1. Neighbor Correction

The main idea of the PISO algorithm is to move the repeated calculations required by SIMPLE and SIMPLEC inside the solution stage of the pressure-correction equation [273] (p. 1068). After one or more additional PISO loops, the corrected velocities satisfy the continuity and momentum equations more closely. This iterative process is called a momentum correction or "neighbor correction". The PISO algorithm takes a little more CPU time per solver iteration, but it can dramatically decrease the number of iterations required for convergence, especially for transient problems.

24.4.3.1.3.2. Skewness Correction

For meshes with some degree of skewness, the approximate relationship between the correction of mass flux at the cell face and the difference of the pressure corrections at the adjacent cells is very rough. Since the components of the pressure-correction gradient along the cell faces are not known in advance, an iterative process similar to the PISO neighbor correction described above is desirable [177] (p. 1063). After the initial solution of the pressure-correction equation, the pressure-correction gradient is recalculated and used to update the mass flux corrections. This process, which is referred to as "skewness correction", significantly reduces convergence difficulties associated with highly distorted meshes. The PISO skewness correction allows Ansys

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Fluent to obtain a solution on a highly skewed mesh in approximately the same number of iterations as required for a more orthogonal mesh. Note that the skewness correction can use a default formulation or an enhanced formulation.

24.4.3.1.3.3. Skewness - Neighbor Coupling

For meshes with a high degree of skewness, the simultaneous coupling of the neighbor and skewness corrections at the same pressure correction equation source may cause divergence or a lack of robustness. An alternate, although more expensive, method for handling the neighbor and skewness corrections inside the PISO algorithm is to apply one or more iterations of skewness correction for each separate iteration of neighbor correction. For each individual iteration of the classical PISO algorithm from [273] (p. 1068), this technique allows a more accurate adjustment of the face mass flux correction according to the normal pressure correction gradient.

24.4.3.1.4. Fractional-Step Method (FSM)

In the FSM, the momentum equations are decoupled from the continuity equation using a mathematical technique called operator-splitting or approximate factorization. The resulting solution algorithm is similar to the segregated solution algorithms described earlier. The formalism used in the approximate factorization allows you to control the order of splitting error. Because of this, the FSM is adopted in Ansys Fluent as a velocity-coupling scheme in a non-iterative time-advancement (NITA) algorithm (Non-Iterative Time-Advancement Scheme (p. 988)).

24.4.3.2. Coupled Algorithm

As previously mentioned, the pressure-based solver allows you to solve your flow problem in either a segregated or coupled manner. Using the coupled approach offers some advantages over the non-coupled or segregated approach. The coupled scheme obtains a robust and efficient single phase implementation for steady-state flows, with superior performance compared to the segregated solution schemes. This pressure-based coupled algorithm offers an alternative to the density-based and pressure-based segregated algorithm with SIMPLE-type pressure-velocity coupling. For transient flows, using the coupled algorithm is necessary when the quality of the mesh is poor, or if large time steps are used.

The pressure-based segregated algorithm solves the momentum equation and pressure correction equations separately. This semi-implicit solution method results in slow convergence.

The coupled algorithm solves the momentum and pressure-based continuity equations together. The full implicit coupling is achieved through an implicit discretization of pressure gradient terms in the momentum equations, and an implicit discretization of the face mass flux, including the Rhie-Chow pressure dissipation terms.

In the momentum equations (Equation 24.41 (p. 977)), the pressure gradient for component k is of the form

$$\sum_f p_f A_k = -\sum_j a^{u_k p} p_j \tag{24.58}$$

Where $a^{u_k p}$ is the coefficient derived from the Gauss divergence theorem and coefficients of the pressure interpolation schemes (Equation 24.42 (p. 977)). Finally, for any ith cell, the discretized form of the momentum equation for component u_k is defined as

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$\sum_j a_{ij}^{u_k u_k} u_{kj} + \sum_j a_{ij}^{u_k p} p_j = b_i^{u_k} \tag{24.59}$$

In the continuity equation, Equation 24.45 (p. 979), the balance of fluxes is replaced using the flux expression in Equation 24.46 (p. 979), resulting in the discretized form

$$\sum_k \sum_j a_{ij}^{pu_k} u_{kj} + \sum_j a_{ij}^{pp} p_j = b_i^{p} \tag{24.60}$$

As a result, the overall system of equations (Equation 24.59 (p. 984) and Equation 24.60 (p. 984)), after being transformed to the δ-form, is presented as

$$\sum_j [A]_{ij} \vec{X}_j = \vec{B}_i \tag{24.61}$$

where the influence of a cell i on a cell j has the form

$$A_{ij} = \begin{bmatrix} a_{ij}^{pp} & a_{ij}^{pu} & a_{ij}^{pv} & a_{ij}^{pw} \\ a_{ij}^{up} & a_{ij}^{uu} & a_{ij}^{uv} & a_{ij}^{uw} \\ a_{ij}^{vp} & a_{ij}^{vu} & a_{ij}^{vv} & a_{ij}^{vw} \\ a_{ij}^{wp} & a_{ij}^{wu} & a_{ij}^{wv} & a_{ij}^{ww} \end{bmatrix} \tag{24.62}$$

and the unknown and residual vectors have the form

$$\vec{X}_j = \begin{bmatrix} p_i' \\ u_i' \\ v_i' \\ w_i' \end{bmatrix} \tag{24.63}$$

$$\vec{B}_i = \begin{bmatrix} -r_i^p \\ -r_i^u \\ -r_i^v \\ -r_i^w \end{bmatrix} \tag{24.64}$$

Note that Equation 24.61 (p. 984) is solved using the coupled AMG, which is detailed in The Coupled and Scalar AMG Solvers (p. 1012).

24.4.3.2.1. Limitation

The pressure-based coupled algorithm is not compatible with the non-iterative time advancement solver (NITA).

24.4.4. Steady-State Iterative Algorithm

If you are performing a steady-state calculation, the governing equations for the pressure-based solver do not contain time-dependent terms. For steady-state flows, Discretization (p. 962) describes control-volume-based discretization of the steady-state transport equation (see Equation 24.1 (p. 961)).

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

24.4.4.1. Under-Relaxation of Variables

The under-relaxation of variables is used in all cases for some material properties, in the NITA solver for solution variables, and in the pressure-based coupled algorithm where this explicit under-relaxation is used for momentum and pressure.

Because of the nonlinearity of the equation set being solved by Ansys Fluent, it is necessary to control the change of φ. This is typically achieved by under-relaxation of variables (also referred to as explicit relaxation), which reduces the change of φ produced during each iteration. In a simple form, the new value of the variable φ within a cell depends upon the old value, φ_{old}, the computed change in φ, $\Delta\varphi$, and the under-relaxation factor, α, as follows:

$$\varphi=\varphi_{old}+\alpha\Delta\varphi \tag{24.65}$$

24.4.4.2. Under-Relaxation of Equations

The under-relaxation of equations, also known as implicit relaxation, is used in the pressure-based solver to stabilize the convergence behavior of the outer nonlinear iterations by introducing selective amounts of φ in the system of discretized equations. This is equivalent to the location-specific time step.

$$\frac{a_p\varphi}{\alpha}=\sum_{nb}a_{nb}\varphi_{nb}+b+\frac{1-\alpha}{\alpha}a_p\varphi_{old} \tag{24.66}$$

The CFL number is a solution parameter in the pressure-based coupled algorithm and can be written in terms of α:

$$\frac{1-\alpha}{\alpha}=\frac{1}{CFL} \tag{24.67}$$

24.4.5. Time-Advancement Algorithm

For time-dependent flows, the discretized form of the generic transport equations is of the following form:

$$\int_V\frac{\partial\rho\varphi}{\partial t}dV+\oint\rho\varphi\vec{v}\cdot d\vec{A}=\oint\Gamma_\varphi\nabla\varphi\cdot d\vec{A}+\int_V S_\varphi dV \tag{24.68}$$

where

$\frac{\partial\rho\varphi}{\partial t}$ = conservative form of transient derivative of transported variable φ

ρ = density

$\vec{v}$ = velocity vector (= $u\hat{i}+v\hat{j}$ in 2D)

$\vec{A}$ = surface area vector

Γ_φ = diffusion coefficient for φ

$\nabla\varphi$ = gradient of φ (= $(\partial\varphi/\partial x)\hat{i}+(\partial\varphi/\partial y)\hat{j}$ in 2D)

S_φ = source of φ per unit volume

The temporal discretization of the transient derivative in the Equation 24.68 (p. 985) is described in Temporal Discretization (p. 968), including first-order and second-order schemes in time. The pressure-

based solver in Ansys Fluent uses an implicit discretization of the transport equation (Equation 24.68 (p. 985)). As a standard default approach, all convective, diffusive, and source terms are evaluated from the fields for time level n+1.

$$\int_V \frac{\partial \rho \varphi}{\partial t} dV + \oint \rho^{n+1} \varphi^{n+1} \vec{v}^{n+1} \cdot d\vec{A} = \oint \Gamma_\varphi{}^{n+1} \nabla \varphi^{n+1} \cdot d\vec{A} + \int_V S_\varphi{}^{n+1} dV \tag{24.69}$$

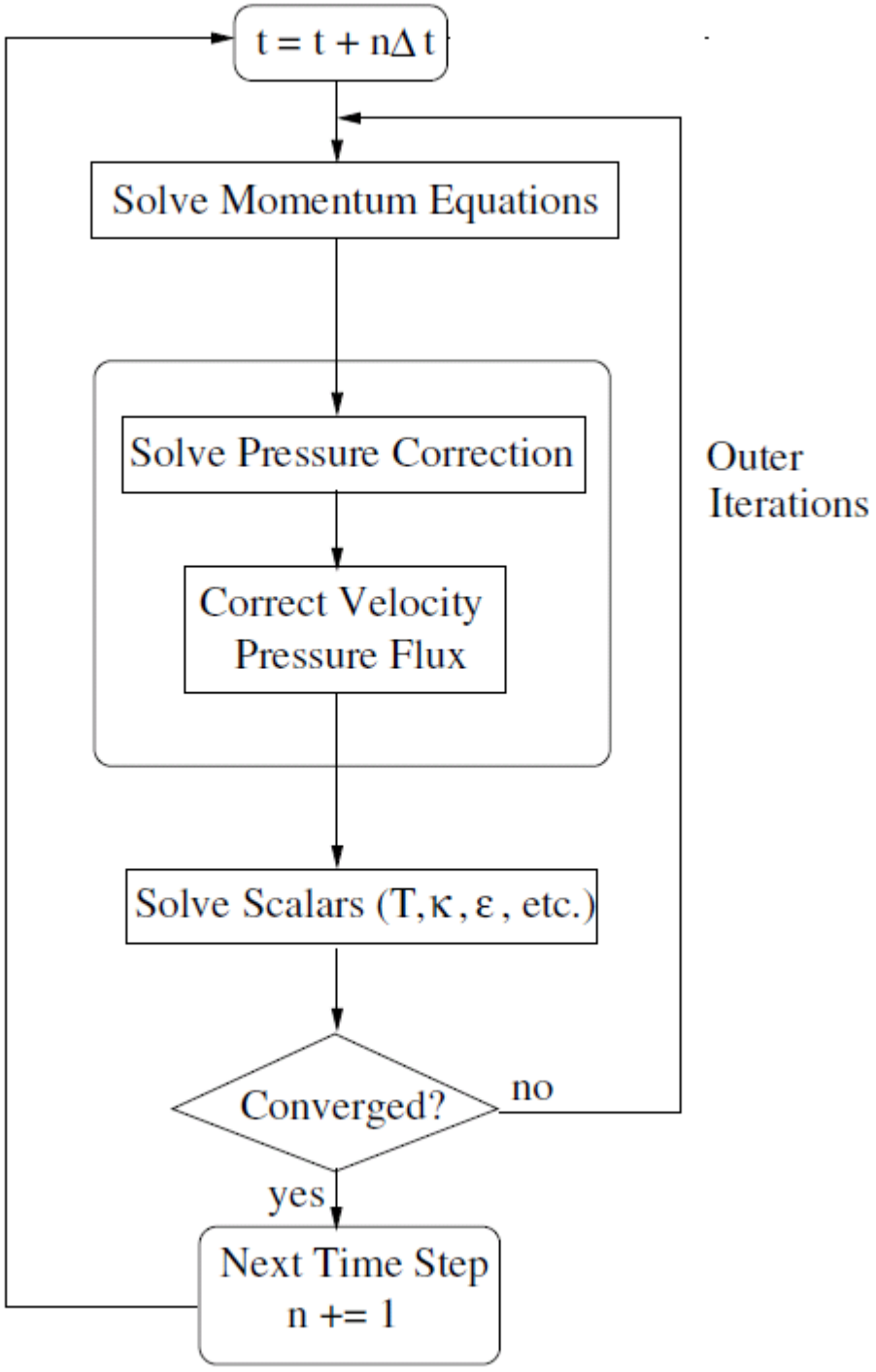
In the pressure-based solver, the overall time-discretization error is determined by both the choice of temporal discretization (for example, first-order, second-order) and the manner in which the solutions are advanced to the next time step (time-advancement scheme). Temporal discretization introduces the corresponding truncation error; $O(\Delta t)$, $O[(\Delta t)^2]$, for first-order and second-order, respectively. The segregated solution process by which the equations are solved one by one introduces splitting error. There are two approaches to the time-advancement scheme depending on how you want to control the splitting error.

24.4.5.1. Iterative Time-Advancement Scheme

In the iterative scheme, all the equations are solved iteratively, for a given time-step, until the convergence criteria are met. Thus, advancing the solutions by one time-step normally requires a number of outer iterations as shown in Figure 24.1: Overview of the Pressure-Based Solution Methods (p. 957) and Figure 24.7: Overview of the Iterative Time Advancement Solution Method For the Segregate Solver (p. 987). With this iterative scheme, non-linearity of the individual equations and inter-equation couplings are fully accounted for, eliminating the splitting error. The iterative scheme is the default in Ansys Fluent.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Figure 24.7: Overview of the Iterative Time Advancement Solution Method For the Segregate Solver



24.4.5.1.1. The Frozen Flux Formulation

The standard fully-implicit discretization of the convective part of Equation 24.69 (p. 986) produces nonlinear terms in the resulting equations. In addition, solving these equations generally requires numerous iterations per time step. As an alternative, Ansys Fluent provides an optional way to

discretize the convective part of Equation 24.68 (p. 985) using the mass flux at the cell faces from the previous time level n.

$$\oint \rho\varphi\vec{v}\cdot d\vec{A} = \oint \rho^{n}\varphi^{n+1}\vec{v}^{n}\cdot d\vec{A} \tag{24.70}$$

The solution still has the same order of accuracy but the nonlinear character of the discretized transport equation is essentially reduced and the convergence within each time step is improved.

The **Frozen Flux Formulation** option is accessible from the **Solution Methods** dialog box.

Important:

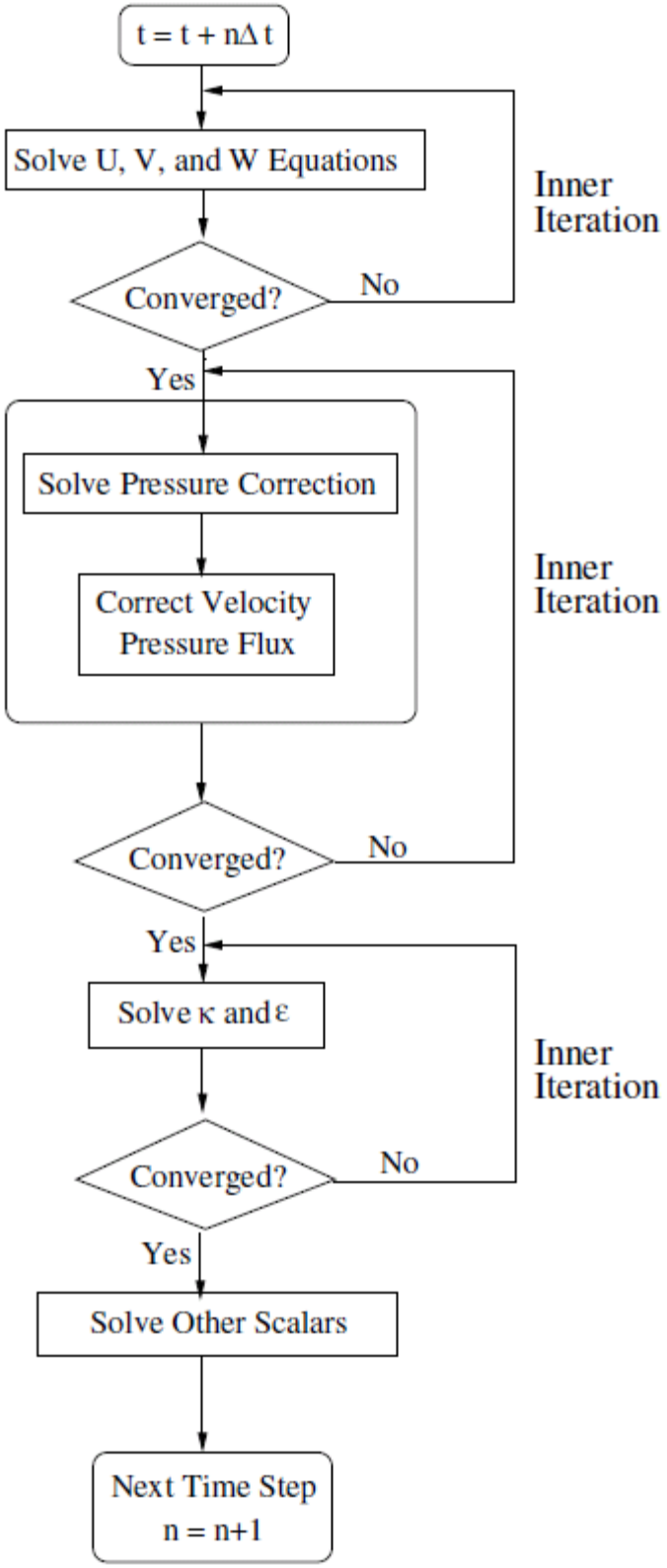
This option is only available for single-phase transient problems that use the pressure-based iterative solver and do not use a moving/deforming mesh model.

24.4.5.2. Non-Iterative Time-Advancement Scheme

The iterative time-advancement scheme requires a considerable amount of computational effort due to a large number of outer iterations performed for each time-step. The idea underlying the non-iterative time-advancement (NITA) scheme is that, in order to preserve overall time accuracy, you do not really need to reduce the splitting error to zero, but only have to make it the same order as the truncation error. The NITA scheme, as seen in Figure 24.8: Overview of the Non-Iterative Time Advancement Solution Method (p. 989), does not need the outer iterations, performing only a single outer iteration per time-step, which significantly speeds up transient simulations. However, the NITA scheme still allows for an inner iteration to solve the individual set of equations.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Figure 24.8: Overview of the Non-Iterative Time Advancement Solution Method



For single phase flows, Ansys Fluent offers two versions of NITA schemes; the non-iterative fractional step method (FSM) ([28] (p. 1054), [153] (p. 1061), [208] (p. 1065), and [209] (p. 1065)) and the non-iterative

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

PISO method [273] (p. 1068). Both NITA schemes are available for first-order and second-order time integration.

For Eulerian multiphase flows, the non-iterative PC Simple scheme is available in Ansys Fluent. Non-iterative PC Simple is a hybrid of the PC Simple and PISO schemes.

Important:

- In general, the NITA solver is not recommended for highly viscous fluid flow.
- The NITA solver is not recommended for flows with a strong dependence of fluid density on species concentration.

24.4.6. Correction Form Discretization of the Momentum Equations

This section describes the background in applying the correction form to the discretized momentum equation.

The correction form of the discretized momentum equation is used by default (and can be disabled using the `solve/set/advanced/correction-form` text command, though this text command will be removed in a future release), as described in Using the Correction Form of Momentum Discretization in the User's Guide.

Discretizing a conservation law, often referred to as a *transport equation*, yields a linear system of equations for the iteration in the following form

$$a_p \varphi_p^n = \sum_{nb} a_{nb} \varphi_{nb}^n + b_p \tag{24.71}$$

Here φ_i are the transported variable cell values, a_{nb} are the coupling coefficients between left and right states, $a_p = \Sigma_{nb} a_{nb}$, and b_p are the cell source terms. Without loss of generality this is written without under-relaxation and transient terms, and is referred to as the *level form*.

By defining the correction from iteration *n-1* to iteration *n* as:

$$\delta\varphi_k^n = \varphi_k^n - \varphi_k^{n-1} \tag{24.72}$$

Equation 24.71 (p. 990) can then be rewritten as

$$a_p \delta\varphi_p^n = \sum_{nb} a_{nb} \delta\varphi_{nb}^n + b_p - \left(a_p \varphi_p^{n-1} - \sum_{nb} a_{nb} \varphi_{nb}^{n-1} \right) \tag{24.73}$$

which is referred to as the *correction* form (or delta form) of the discretized equation.

24.5. Density-Based Solver

The density-based solver in Ansys Fluent solves the governing equations of continuity, momentum, and (where appropriate) energy and species transport simultaneously as a set, or vector, of equations. Governing equations for additional scalars will be solved sequentially (that is, segregated from one another and from the coupled set), in a manner described in General Scalar Transport Equation: Discretization and Solution (p. 960). Two algorithms are available for solving the coupled set of equations, the coupled-explicit formulation and the coupled-implicit formulation.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Information is organized into the following subsections:

24.5.1. Governing Equations in Vector Form
24.5.2. Preconditioning
24.5.3. Convective Fluxes
24.5.4. Steady-State Flow Solution Methods
24.5.5. Unsteady Flows Solution Methods

24.5.1. Governing Equations in Vector Form

The system of governing equations for a single-component fluid, written to describe the mean flow properties, is cast in integral Cartesian form for an arbitrary control volume V with differential surface area dA as follows:

$$\frac{\partial}{\partial t}\int_V W dV + \oint [F - G] \cdot dA = \int_V H dV \tag{24.74}$$

where the vectors W, F, and G are defined as

$$W = \begin{Bmatrix} \rho \\ \rho u \\ \rho v \\ \rho w \\ \rho E \end{Bmatrix}, F = \begin{Bmatrix} \rho v \\ \rho v u + p\hat{i} \\ \rho v v + p\hat{j} \\ \rho v w + p\hat{k} \\ \rho v E + pv \end{Bmatrix}, G = \begin{Bmatrix} 0 \\ \tau_{xi} \\ \tau_{yi} \\ \tau_{zi} \\ \tau_{ij} v_j + q \end{Bmatrix} \tag{24.75}$$

and the vector H contains source terms such as body forces and energy sources.

Here ρ, v, E, and p are the density, velocity, total energy per unit mass, and pressure of the fluid, respectively. τ is the viscous stress tensor, and q is the heat flux.

Total energy E is related to the total enthalpy H by

$$E = H - p/\rho \tag{24.76}$$

where

$$H = h + |v|^2/2 \tag{24.77}$$

The Navier-Stokes equations as expressed in Equation 24.74 (p. 991) become (numerically) very stiff at low Mach number due to the disparity between the fluid velocity v and the acoustic speed c (speed of sound). This is also true for incompressible flows, regardless of the fluid velocity, because acoustic waves travel infinitely fast in an incompressible fluid (speed of sound is infinite). The numerical stiffness of the equations under these conditions results in poor convergence rates. This difficulty is overcome in Ansys Fluent's density-based solver by employing a technique called (time-derivative) preconditioning [695] (p. 1093).

24.5.2. Preconditioning

Time-derivative preconditioning modifies the time-derivative term in Equation 24.74 (p. 991) by pre-multiplying it with a preconditioning matrix. This has the effect of re-scaling the acoustic speed (eigenvalue) of the system of equations being solved in order to alleviate the numerical stiffness encountered in low Mach numbers and incompressible flow.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Derivation of the preconditioning matrix begins by transforming the dependent variable in Equation 24.74 (p. 991) from conserved quantities W to primitive variables Q using the chain-rule as follows:

$$\frac{\partial W}{\partial Q}\frac{\partial}{\partial t}\int_V Q dV + \oint [F - G]\cdot dA = \int_V H dV \tag{24.78}$$

where Q is the vector $\{p, u, v, w, T\}^T$ and the Jacobian $\partial W / \partial Q$ is given by

$$\frac{\partial W}{\partial Q} = \begin{bmatrix} \rho_p & 0 & 0 & 0 & \rho_T \\ \rho_p u & \rho & 0 & 0 & \rho_T u \\ \rho_p v & 0 & \rho & 0 & \rho_T v \\ \rho_p w & 0 & 0 & \rho & \rho_T w \\ \rho_p H - \delta & \rho u & \rho v & \rho w & \rho_T H + \rho C_p \end{bmatrix} \tag{24.79}$$

where

$$\rho_p = \left.\frac{\partial \rho}{\partial p}\right|_T, \rho_T = \left.\frac{\partial \rho}{\partial T}\right|_p \tag{24.80}$$

and $\delta = 1$ for an ideal gas and $\delta = 0$ for an incompressible fluid.

The choice of primitive variables Q as dependent variables is desirable for several reasons. First, it is a natural choice when solving incompressible flows. Second, when we use second-order accuracy we need to reconstruct Q rather than W in order to obtain more accurate velocity and temperature gradients in viscous fluxes, and pressure gradients in inviscid fluxes. And finally, the choice of pressure as a dependent variable allows the propagation of acoustic waves in the system to be singled out [667] (p. 1091).

We precondition the system by replacing the Jacobian matrix $\partial W / \partial Q$ (Equation 24.79 (p. 992)) with the preconditioning matrix Γ so that the preconditioned system in conservation form becomes

$$\Gamma\frac{\partial}{\partial t}\int_V Q dV + \oint [F - G]\cdot dA = \int_V H dV \tag{24.81}$$

where

$$\Gamma = \begin{bmatrix} \Theta & 0 & 0 & 0 & \rho_T \\ \Theta u & \rho & 0 & 0 & \rho_T u \\ \Theta v & 0 & \rho & 0 & \rho_T v \\ \Theta w & 0 & 0 & \rho & \rho_T w \\ \Theta H - \delta & \rho u & \rho v & \rho w & \rho_T H + \rho C_p \end{bmatrix} \tag{24.82}$$

The parameter Θ is given by

$$\Theta = \left(\frac{1}{U_r^2} - \frac{\rho_T}{\rho C_p}\right) \tag{24.83}$$

The reference velocity U_r appearing in Equation 24.83 (p. 992) is chosen locally such that the eigenvalues of the system remain well conditioned with respect to the convective and diffusive time scales [695] (p. 1093).

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

The resultant eigenvalues of the preconditioned system (Equation 24.81 (p. 992)) are given by

$$u, u, u, u'+c', u'-c' \tag{24.84}$$

where

$$u = v \cdot \hat{n}$$
$$u' = u(1-\alpha)$$
$$c' = \sqrt{\alpha^2 u^2 + U_r^2}$$
$$\alpha = \left(1-\beta U_r^2\right) / 2$$
$$\beta = \left(\rho_p + \frac{\rho_T}{\rho C_p}\right)$$

For an ideal gas, $\beta = \left(\gamma RT\right)^{-1} = 1/c^2$. Thus, when $U_r = c$ (at sonic speeds and above), $\alpha = 0$ and the eigenvalues of the preconditioned system take their traditional form, $u \pm c$. At low speed, however, as $U_r \to 0$, $\alpha \to 1/2$ and all eigenvalues become of the same order as u. For constant-density flows, $\beta = 0$ and $\alpha = 1/2$ regardless of the values of U_r. As long as the reference velocity is of the same order as the local velocity, all eigenvalues remain of the order u. Thus, the eigenvalues of the preconditioned system remain well conditioned at all speeds.

Note that the non-preconditioned Navier-Stokes equations are recovered exactly from Equation 24.81 (p. 992) by setting $1/U_r^2$ to ρ_p, the derivative of density with respect to pressure. In this case Γ reduces exactly to the Jacobian $\partial W / \partial Q$.

Although Equation 24.81 (p. 992) is conservative in the steady state, it is not, in a strict sense, conservative for time-dependent flows. This is not a problem, however, since the preconditioning has already destroyed the time accuracy of the equations and we will not employ them in this form for unsteady calculations.

For unsteady calculations, an unsteady preconditioning is available when the dual-time stepping method is used (Implicit Time Stepping (Dual-Time Formulation) (p. 998)). The unsteady preconditioning enhances the solution accuracy by improving the scaling of the artificial dissipation and maximizes the efficiency by optimizing the number of sub-iterations required at each physical time step [501] (p. 1082). For low Mach number flows in particular, for both low frequency problems (large time steps) and high frequency problems (small time step), significant savings in computational time are possible when compared with the non-preconditioned case.

The unsteady preconditioning adapts the level of preconditioning based on the user-specified time-step and on the local advective and acoustic time scales of the flow. For acoustic problems, the physical time-step size is small as it is based on the acoustic CFL number. In this case the preconditioning parameter U_r^2 will approach c^2, which in effect will turn off the low-Mach preconditioning almost completely. For advection dominated problems, like the transport of turbulent vortical structures, and so on, the physical time-step is large as it is based on the particle CFL number. The corresponding unsteady preconditioning parameter U_r^2 will then approach u^2, which corresponds to the steady preconditioning choice. For intermediate physical time-step sizes, the unsteady preconditioning parameter will be adapted to provide optimum convergence efficiency of the pseudo-time iterations and accurate scaling of the artificial dissipation terms, regardless of the choice of the physical time step.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

24.5.3. Convective Fluxes

24.5.3.1. Roe Flux-Difference Splitting Scheme

The inviscid flux vector F appearing in Equation 24.81 (p. 992) is evaluated by a standard upwind, flux-difference splitting [556] (p. 1085). This approach acknowledges that the flux vector F contains characteristic information propagating through the domain with speed and direction according to the eigenvalues of the system. By splitting F into parts, where each part contains information traveling in a particular direction (that is, characteristic information), and upwind differencing the split fluxes in a manner consistent with their corresponding eigenvalues, we obtain the following expression for the discrete flux at each face:

$$F=\frac{1}{2}\left(F_R+F_L\right)-\frac{1}{2}\Gamma\left|\hat{A}\right|\delta Q \tag{24.85}$$

Here δQ is the spatial difference Q_R-Q_L. The fluxes $F_R=F\left(Q_R\right)$ and $F_L=F\left(Q_L\right)$ are computed using the (reconstructed) solution vectors Q_R and Q_L on the "right" and "left" side of the face. The matrix $\left|\hat{A}\right|$ is defined by

$$\left|\hat{A}\right|=M\left|\Lambda\right|M^{-1} \tag{24.86}$$

where Λ is the diagonal matrix of eigenvalues and M is the modal matrix that diagonalizes Γ^{-1}A, where A is the inviscid flux Jacobian $\partial F/\partial Q$.

For the non-preconditioned system (and an ideal gas) Equation 24.85 (p. 994) reduces to Roe's flux-difference splitting [556] (p. 1085) when Roe-averaged values are used to evaluate $\Gamma\left|\hat{A}\right|$. At present, arithmetic averaging of states Q_R and Q_L is used.

In its current form, Equation 24.85 (p. 994) can be viewed as a second-order central difference plus an added matrix dissipation. The added matrix dissipation term is not only responsible for producing an upwinding of the convected variables, and of pressure and flux velocity in supersonic flow, but it also provides the pressure-velocity coupling required for stability and efficient convergence of low-speed and incompressible flows.

24.5.3.2. AUSM+ Scheme

An alternative way to compute the flux vector F appearing in Equation 24.81 (p. 992) is by using a flux-vector splitting scheme [107] (p. 1059). The scheme, called Advection Upstream Splitting Method (AUSM), was first introduced by Liou and Steffen in 1993 [378] (p. 1075). The AUSM scheme first computes a cell interface Mach number based on the characteristic speeds from the neighboring cells. The interface Mach number is then used to determine the upwind extrapolation for the convection part of the inviscid fluxes. A separate Mach number splitting is used for the pressure terms. Generalized Mach number based convection and pressure splitting functions were proposed by Liou [377] (p. 1074) and the new scheme was termed AUSM+. The AUSM+ scheme has several desirable properties:

1. Provides exact resolution of contact and shock discontinuities
2. Preserves positivity of scalar quantities
3. Free of oscillations at stationary and moving shocks

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

The AUSM+ scheme avoids using an explicit artificial dissipation, by proposing a numerical flux of the form:

$$F = m_f \varphi + p_i \tag{24.87}$$

Here m_f is the mass flux through the interface, which is computed using the fourth order polynomial functions of the left and right side (of the interface) Mach numbers.

Ansys Fluent utilizes an all-speed version of the AUSM+ scheme based on the low-Mach preconditioning.

24.5.3.3. Low Diffusion Roe Flux Difference Splitting Scheme

In order to reduce dissipation in LES calculations, Ansys Fluent uses a modified version of the Roe Flux Difference Splitting scheme, called the Low Diffusion Roe Flux Difference Splitting scheme. The scheme includes low Mach number preconditioning, in which the artificial dissipation term has been reduced [87] (p. 1058) through the use of a hybrid scheme that combines a central scheme and a second-order upwind scheme (Roe's Flux Difference scheme).

The Low Diffusion Roe-FDS is available in special circumstances when the LES viscous model is selected and when using the implicit-time formulation (dual-time-stepping). It is recommended that you use the second-order upwind scheme for maximum accuracy when selecting this flux scheme.

Important:

The low diffusion discretization must be used only for subsonic flows. For high Mach number flows, it is recommended that you switch to standard Roe-FDS or AUSM+ flux schemes and second-order upwind discretization.

Important:

When running LES with the explicit-time formulation, Low Diffusion Roe-FDS is not available and you will need to either use the standard Roe-FDS or the AUSM+ flux scheme and second-order upwind discretization.

To learn how to apply the different convective fluxes, see Convective Flux Types in the User's Guide.

24.5.4. Steady-State Flow Solution Methods

The coupled set of governing equations (Equation 24.81 (p. 992)) in Ansys Fluent is discretized in time for both steady and unsteady calculations. In the steady case, it is assumed that time marching proceeds until a steady-state solution is reached. Temporal discretization of the coupled equations is accomplished by either an implicit or an explicit time-marching algorithm. These two algorithms are described below. To learn how to apply the two formulations, see Density-Based Solver Settings in the User's Guide.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

24.5.4.1. Explicit Formulation

In the explicit scheme a multi-stage, time-stepping algorithm [276] (p. 1069) is used to discretize the time derivative in Equation 24.81 (p. 992). The solution is advanced from iteration n to iteration $n+1$ with an m-stage Runge-Kutta scheme, given by

$$Q^0 = Q^n$$
$$\Delta Q^i = -\alpha_i \Delta t \Gamma^{-1} R^{i-1}$$
$$Q^{n+1} = Q^m$$

where $\Delta Q^i \equiv Q^i - Q^n$ and $i = 1, 2, \ldots, m$ is the stage counter for the m-stage scheme. α_i is the multi-stage coefficient for the i^{th} stage. The residual R^i is computed from the intermediate solution Q^i and, for Equation 24.81 (p. 992), is given by

$$R^i = \sum^{N_{faces}} \left(F\left(Q^i\right) - G\left(Q^i\right) \right) \cdot A - VH \tag{24.88}$$

The time step Δt is computed from the CFL (Courant-Friedrichs-Lewy) condition

$$\Delta t = \frac{2 \ CFL \ V}{\sum_f \lambda_f^{\max} A_f} \tag{24.89}$$

where V is the cell volume, A_f is the face area, and λ_f^{max} is the maximum of the local eigenvalues defined by Equation 24.84 (p. 993).

For steady-state solutions, convergence acceleration of the explicit formulation can be achieved with the use of local time stepping, residual smoothing, and full-approximation storage multigrid.

Local time stepping is a method by which the solution at each control volume is advanced in time with respect to the cell time step, defined by the local stability limit of the time-stepping scheme.

Residual smoothing, on the other hand, increases the bound of stability limits of the time-stepping scheme and hence allows for the use of a larger CFL value to achieve fast convergence (Implicit Residual Smoothing (p. 996)).

The convergence rate of the explicit scheme can be accelerated through use of the full-approximation storage (FAS) multigrid method described in Full-Approximation Storage (FAS) Multigrid (p. 1014).

By default, Ansys Fluent uses a 3-stage Runge-Kutta scheme based on the work by Lynn [399] (p. 1076) for steady-state flows that use the density-based explicit solver.

24.5.4.1.1. Implicit Residual Smoothing

The maximum time step can be further increased by increasing the support of the scheme through implicit averaging of the residuals with their neighbors. The residuals are filtered through a Laplacian smoothing operator:

$$\bar{R}_i = R_i + \varepsilon \sum \left(\bar{R}_j - \bar{R}_i \right) \tag{24.90}$$

This equation can be solved with the following Jacobi iteration:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$\bar{R}_i^m = \frac{R_i + \epsilon \sum_N^{} \bar{R}_j^{m-1}}{1 + \epsilon N} \tag{24.91}$$

where N is the number of neighbors. Two Jacobi iterations are usually sufficient to allow doubling the time step with a value of $\varepsilon = 0.5$.

24.5.4.2. Implicit Formulation

In the implicit scheme, an Euler implicit discretization in time of the governing equations (Equation 24.81 (p. 992)) is combined with a Newton-type linearization of the fluxes to produce the following linearized system in delta form [693] (p. 1093):

$$\left[D + \sum_j^{N_{faces}} S_{j,k} \right] \Delta Q^{n+1} = -R^n \tag{24.92}$$

The center and off-diagonal coefficient matrices, D and $S_{j,k}$ are given by

$$D = \frac{V}{\Delta t} \Gamma + \sum_j^{N_{faces}} S_{j,i} \tag{24.93}$$

$$S_{j,k} = \left(\frac{\partial F_j}{\partial Q_k} - \frac{\partial G_j}{\partial Q_k} \right) A_j \tag{24.94}$$

and the residual vector R^n and time step Δt are defined as in Equation 24.88 (p. 996) and Equation 24.89 (p. 996), respectively.

Equation 24.92 (p. 997) is solved using either Incomplete Lower Upper factorization (ILU) by default or symmetric point Gauss-Seidel algorithm, in conjunction with an algebraic multigrid (AMG) method (see Algebraic Multigrid (AMG) (p. 1009)) adapted for coupled sets of equations.

Explicit relaxation can improve the convergence to steady state of the implicit formulation. By default, explicit relaxation is enabled for the implicit solver and uses a factor of `0.75`. You can specify a factor α to control the amount that the solution vector Q changes between iterations after the end of the algebraic multigrid (AMG) cycle:

$$Q_{new} = Q_{old} + \alpha \Delta Q \tag{24.95}$$

By specifying a value less than the default value of `1` for α, the variables in the solution vector will be under-relaxed and the convergence history can be improved. For information on how to set this value, see Specifying the Explicit Relaxation in the User's Guide.

Important:

Note that explicit relaxation is available for the density-based implicit solver in steady state mode only.

24.5.4.2.1. Convergence Acceleration for Stretched Meshes

When the flow is solved on meshes containing highly stretched cells (cells with large aspect ratio typically found near wall boundaries), the traditional implicit solution method will suffer from convergence slow down due to the small local time step values defined by equation Equa-

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

tion 24.89 (p. 996). The larger the cell aspect ratio, the smaller the local time step and the slower the local rate change of the solution. The time step definition in equation Equation 24.89 (p. 996) is based effectively on the minimum cell characteristic distance. A more optimum time step definition for the implicit solution method can be based on the maximum characteristic length of the cell [362] (p. 1074) to give a new time step definition:

$$\Delta t = \frac{2 \quad CFL \quad V \quad AR}{\sum_f \lambda_f^{\max} A_f} \tag{24.96}$$

where AR is the cell aspect ratio defined as the ratio of maximum length to minimum length in the cell $AR = \frac{l_{\max}}{l_{\min}}$

Effectively, the new definition will vary the local CFL value from one cell to another proportional to the value of the cell aspect ratio. When the cell aspect ratio is near unity, the local CFL value will be similar to the one entered by you. On the other hand, on stretched high aspect ratio cells, the local CFL value is a multiple of the cell aspect ratio. The new time step definition will accelerate the solution convergence particularly on meshes with Y+ near 1.

24.5.5. Unsteady Flows Solution Methods

For time-accurate calculations, explicit and implicit time-stepping schemes are available. (The time-implicit approach is also referred to as "dual time stepping".)

24.5.5.1. Explicit Time Stepping

The explicit time stepping approach, is available only for the explicit scheme described in Explicit Formulation (p. 996). The time step is determined by the CFL condition. To maintain time accuracy of the solution the explicit time stepping employs the same time step in each cell of the domain (this is also known as global-time step), and with preconditioning disabled. By default, Ansys Fluent uses a 4-stage Runge-Kutta scheme for unsteady flows.

24.5.5.2. Implicit Time Stepping (Dual-Time Formulation)

The implicit-time stepping method (also known as dual-time formulation) is available in the density-based explicit and implicit formulation.

When performing unsteady simulations with implicit-time stepping (dual-time stepping), Ansys Fluent uses a low Mach number time-derivative unsteady preconditioner to provide accurate results both for pure convective processes (for example, simulating unsteady turbulence) and for acoustic processes (for example, simulating wave propagation) [658] (p. 1091), [501] (p. 1082).

Here we introduce a preconditioned pseudo-time-derivative term into Equation 24.74 (p. 991) as follows:

$$\frac{\partial}{\partial t}\int_V W dV + \Gamma \frac{\partial}{\partial \tau}\int_V Q dV + \oint [F - G] \cdot dA = \int_V H dV \tag{24.97}$$

where t denotes physical-time and τ is a pseudo-time used in the time-marching procedure. Note that as $\tau \to \infty$, the second term on the left side of Equation 24.97 (p. 998) vanishes and Equation 24.74 (p. 991) is recovered.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

The time-dependent term in Equation 24.97 (p. 998) is discretized in an implicit fashion by means of either a first- or second-order accurate, backward difference in time.

The dual-time formulation is written in semi-discrete form as follows:

$$\left[\frac{\Gamma}{\Delta\tau}+\frac{\varepsilon_0}{\Delta t}\frac{\partial W}{\partial Q}\right]\Delta Q^{k+1}+\frac{1}{V}\oint[F-G]\cdot dA=H-\frac{1}{\Delta t}\left(\varepsilon_0 W^k-\varepsilon_1 W^n+\varepsilon_2 W^{n-1}\right) \tag{24.98}$$

where $\{\varepsilon_0=\varepsilon_1=1/2,\varepsilon_2=0\}$ gives first-order time accuracy, and $\{\varepsilon_0=3/2,\varepsilon_1=2,\varepsilon_2=1/2\}$ gives second-order time accuracy (note that this is for the fixed time step size formulation; for the variable time step size formulation, see Equation 24.27 (p. 971)). k is the inner iteration counter and n represents any given physical-time level.

The pseudo-time-derivative is driven to zero at each physical time level by a series of inner iterations using either the implicit or explicit time-marching algorithm.

Throughout the (inner) iterations in pseudo-time, W^n and W^{n-1} are held constant and W^k is computed from Q^k. As $\tau\to\infty$, the solution at the next physical time level W^{n+1} is given by $W\left(Q^k\right)$.

Note that the physical time step size Δt is limited only by the level of desired temporal accuracy. The pseudo time step size $\Delta\tau$ is determined by the CFL condition of the (implicit or explicit) time-marching scheme.

Table 24.1: Summary of the Density-Based Solver (p. 999) summarizes all operation modes for the density-based solver from the iterative scheme in steady-state calculations to time-marching schemes for transient calculations.

Table 24.1: Summary of the Density-Based Solver

Solution Method	Density-Based Solver - Explicit Formulation	Density-Based Solver - Implicit Formulation
Steady-State	– 3-stage Runge-Kutta	– local time step
	– local time step	– time-derivative preconditioning
	– time-derivative preconditioning	
	— FAS	
Unsteady - Explicit Time Stepping	– 4-stage Runge-Kutta	N/A
	– global time step	
	– no time-derivative preconditioning	
	– No FAS	
Unsteady - Implicit Time Stepping (dual-time formulation) First Order	– dual-time formulation	– dual-time formulation
	– Physical time: first order Euler backward	– Physical time: first order Euler backward
	– preconditioned pseudo-time derivative	– preconditioned pseudo-time derivative
	– inner iteration: explicit pseudo-time marching, 3-stage Runge-Kutta	– inner iteration: implicit pseudo-time marching
Unsteady - Implicit Time	– dual-time formulation	– dual-time formulation

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Solution Method	Density-Based Solver - Explicit Formulation	Density-Based Solver - Implicit Formulation
Stepping (dual-time formulation) Second Order	– Physical time: second-order Euler backward	– Physical time: second-order Euler backward
	– preconditioned pseudo-time derivative	– preconditioned pseudo-time derivative
	– inner iteration: explicit pseudo-time marching, 3-stage Runge-Kutta	– inner iteration: implicit pseudo-time marching

24.6. Pseudo Time Method Under-Relaxation

When a pseudo time method is selected, an advanced form of implicit under-relaxation is applied that adjusts the relaxation factor dynamically during the simulation according to the flow field behavior. After introducing the pseudo time method into the generic transport equation, the integral form of governing equation is of the following forms:

- For steady-state calculations:

$$\int_V \frac{\partial \rho\phi}{\partial \tau} dV + \oint_S \rho\phi\vec{v}\cdot d\vec{A} = \oint_S \Gamma_\phi \nabla\phi \cdot d\vec{A} + \int_V S_\phi dV \tag{24.99}$$

- For transient calculations:

$$\int_V \frac{\partial \rho\phi}{\partial t} dV + \int_V \frac{\partial \rho\phi}{\partial \tau} dV + \oint_S \rho\phi\vec{v}\cdot d\vec{A} = \oint_S \Gamma_\phi \nabla\phi \cdot d\vec{A} + \int_V S_\phi dV \tag{24.100}$$

where τ denotes the pseudo time. Note that the pseudo time terms vanish and the original form of the equations is recovered when $\tau \to \infty$.

After discretizating the governing equation using the finite volume method, the algebraic form of the steady-state equation is obtained as

$$\rho_p \Delta V \frac{\phi_p - \phi_p^{\text{old}}}{\Delta\tau} + a_p\phi_p = \sum_{nb} a_{nb}\phi_{nb} + b \tag{24.101}$$

where ϕ_p^{old} denotes the value of ϕ_p at the previous iteration, and $\Delta\tau$ is the pseudo time step size that can be computed using the local or global time step method (as described in the sections that follow). For the local time step method, Equation 24.101 (p. 1000) is extended to transient flow in a straightforward manner.

To learn how to apply under-relaxation using the pseudo time methods, refer to Performing Calculations with a Pseudo Time Method.

For more information, see the following sections:

24.6.1. Local Time Step Method

24.6.2. Global Time Step Method

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

24.6.1. Local Time Step Method

The local time step formulation for the pseudo time method is compatible with steady-state and transient cases that use a pressure-based segregated solver (SIMPLE, SIMPLEC, or PISO). Fluent computes a local pseudo time step size for each cell in the domain based your specified pseudo time Courant number and the flow scales [501] (p. 1082). In addition to the under-relaxation implied by the pseudo time method, implicit under-relaxation is applied by default for the momentum equation and other transport equations.

For details on applying the local time step method, refer to Local Time Step Method Setting in the User's Guide.

24.6.2. Global Time Step Method

The global time step formulation for the pseudo time method is only compatible with steady-state cases that use the pressure-based coupled solver or the density-based implicit solver. It uses a single pseudo time step size for the whole simulation domain. You have the option of specifying the pseudo time step size or you can have Ansys Fluent calculate it automatically; the automatically calculated value can be further modified if you specify a time scale factor other than 1. Ansys Fluent allows you to use a different time step size for the fluid domain and the following solid domains:

- Solid zones
- Porous zones
- Solidification and melting model

The automatic time step is calculated as follows:

- *Fluid Time Scale*

  The automatic fluid time scale for fluid zones ($\Delta\tau_{\text{fluid}}$) is calculated using the minimum of the different time scales:

  $$\Delta\tau_{\text{fluid}}=\text{Min}\left(\Delta\tau_{\text{u}},\Delta\tau_{\text{p}},\Delta\tau_{\text{b}},\Delta\tau_{\text{g}},\Delta\tau_{\text{rot}},\Delta\tau_{\text{compress}}\right) \tag{24.102}$$

  where $\Delta\tau_{\text{u}}$, $\Delta\tau_{\text{p}}$, $\Delta\tau_{\text{b}}$, $\Delta\tau_{\text{g}}$, $\Delta\tau_{\text{rot}}$, and $\Delta\tau_{\text{compress}}$ represent the convection, dynamic, buoyancy, gravitational, rotational, and compressible time scales.

  Depending on the physics of the problem and applicability of the previously mentioned time scales, they may be used for computing the global pseudo time step size. Each time scale can be obtained by dividing a representative velocity scale by a representative length scale L_{scale}.

  The convective time scale $\Delta\tau_{\text{u}}$ is calculated as:

  $$\Delta\tau_{\text{u}}=\frac{0.3L_{\text{scale}}}{\text{Max}\left(0.1U_{\text{bc}},U_{\text{domain}}\right)} \tag{24.103}$$

  where the velocity U_{bc} is the maximum of the arithmetic average of the velocity at the domain boundary faces, and U_{domain} is the arithmetic average of the velocity over the cells in the domain. For the VOF and mixture models, the average velocity is calculated using the velocity in interfacial cells only.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

The dynamic time scale $\Delta\tau_{\mathrm{p}}$ is calculated as:

$$\Delta\tau_{\mathrm{p}}=\frac{0.3L_{\mathrm{scale}}}{U_{\Delta P}} \tag{24.104}$$

where the velocity scale $U_{\Delta P}$ is based on the pressure difference at open boundaries, such as pressure inlet, pressure outlet or velocity inlet, and is defined by the following:

$$U_{\Delta P}=0.1\sqrt{\frac{\left(P_{\mathrm{bc,max}}-P_{\mathrm{bc,min}}\right)}{\bar{\rho}_{\mathrm{cells}}}} \tag{24.105}$$

where $P_{\mathrm{bc,max}}$ is the maximum pressure value at the open boundary, $P_{\mathrm{bc,min}}$ is the minimum pressure value at the open boundary, and $\bar{\rho}_{\mathrm{cells}}$ is the average density over the domain.

When using the Boussinesq buoyancy model, the buoyancy time scale $\Delta\tau_{\mathrm{b}}$ is calculated as:

$$\Delta\tau_{\mathrm{b}}=\sqrt{\frac{L_{\mathrm{scale}}}{g\beta\left(T_{\mathrm{max}}-T_{\mathrm{min}}\right)}} \tag{24.106}$$

where g is the magnitude of the gravitational vector (that is, $g=\left|\vec{g}\right|$), β is the thermal expansion coefficient, and T_{max} and T_{min} are the maximum and minimum temperatures, respectively.

For the full (non-Boussinesq) buoyancy model, the gravitational time scale $\Delta\tau_{\mathrm{g}}$ is calculated as:

$$\Delta\tau_{\mathrm{g}}=\sqrt{\frac{\rho L_{\mathrm{scale}}}{g\left(\rho_{\mathrm{max}}-\rho_{\mathrm{min}}\right)}} \tag{24.107}$$

where ρ_{max} and ρ_{min} are the maximum and minimum densities, respectively, and ρ is the average density.

For cases with rotational velocities, the rotational time scale $\Delta\tau_{\mathrm{rot}}$ is calculated as

$$\Delta\tau_{\mathrm{rot}}=\frac{0.1}{\omega_{\mathrm{max}}} \tag{24.108}$$

where ω_{max} is the maximum of the magnitude of the angular velocity vector (that is, $\omega_{\mathrm{max}}=\mathrm{Max}\left(\left|\vec{\omega}\right|\right)$).

If the simulation Mach number M is greater than 0.3, then the compressible time scale $\Delta\tau_{\mathrm{compress}}$ is obtained from the following formula:

$$\Delta\tau_{\mathrm{compress}}=\frac{0.3L_{\mathrm{scale}}}{\overline{c}} \tag{24.109}$$

The simulation Mach number M is computed as

$$M=\frac{\mathrm{Max}\left(0.1U_{\mathrm{bc}},U_{\mathrm{domain}},U_{\Delta P}\right)}{\overline{c}} \tag{24.110}$$

where $\overline{c}$ is the average speed of sound over all cells.

Two length scale calculation methods are available: *aggressive* and *conservative*. If the aggressive length scale method is applied, then the length scale in all of the equations shown previously are defined as

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$L_{\text{scale}} = \text{Max}\left(L_{\text{vol}}, L_{\text{ext}}\right) \tag{24.111}$$

On the other hand, if the *conservative* length scale method is applied, then the length scale is defined as

$$L_{\text{scale}} = \text{Min}\left(L_{\text{vol}}, L_{\text{ext}}\right) \tag{24.112}$$

where the volumetric length scale is defined as

$$L_{\text{vol}} = \sqrt[3]{V} \tag{24.113}$$

and the domain length scale is defined as

$$L_{\text{ext}} = \text{Max}\left(L_x, L_y, L_z\right) \tag{24.114}$$

where V is the volume of the domain and (L_x, L_y, L_z) are the domain extents in the x, y, and z direction, respectively.

- *Solid Time Scale*

  The automatic solid time scale for solid zones, porous zones, and the solidification and melting model is assumed to be infinite (1e20 s); however, this value can be reduced by reducing the time scale factor for the solid zones.

For details on applying the automatic time step for the global time step method, refer to Global Time Step Method Settings in the User's Guide.

24.7. Multigrid Method

The Ansys Fluent solver contains two forms of multigrid: algebraic (AMG) and full-approximation storage (FAS). AMG is an essential component of both the pressure-based and density-based implicit solvers, while FAS is an important, but optional, component of the density-based explicit solver. (Note that when the density-based explicit solver is used, AMG will also be used, since the scalar equations (for example, turbulence) are solved using the approach described in General Scalar Transport Equation: Discretization and Solution (p. 960).)

This section describes the mathematical basis of the multigrid approach. Common aspects of AMG and FAS are presented first, followed by separate sections that provide details unique to each method.

Information is organized into the following subsections:

24.7.1. Approach

24.7.2. Multigrid Cycles

24.7.3. Algebraic Multigrid (AMG)

24.7.4. Full-Approximation Storage (FAS) Multigrid

For information about user inputs and controls for the multigrid solver, see Modifying Algebraic Multigrid Parameters and Setting FAS Multigrid Parameters in the User's Guide.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

24.7.1. Approach

Ansys Fluent uses a multigrid scheme to accelerate the convergence of the solver by computing corrections on a series of coarse grid levels. The use of this multigrid scheme can greatly reduce the number of iterations and the CPU time required to obtain a converged solution, particularly when your model contains a large number of control volumes.

24.7.1.1. The Need for Multigrid

Implicit solution of the linearized equations on unstructured meshes is complicated by the fact that there is no equivalent of the line-iterative methods that are commonly used on structured meshes. Since direct matrix inversion is out of the question for realistic problems and "whole-field" solvers that rely on conjugate-gradient (CG) methods have robustness problems associated with them, the methods of choice are point implicit solvers like Gauss-Seidel and ILU. Although the Gauss-Seidel and ILU schemes rapidly remove local (high-frequency) errors in the solution, global (low-frequency) errors are reduced at a rate inversely related to the mesh size. Thus, for a large number of nodes, the solver "stalls" and the residual reduction rate becomes prohibitively low.

The multi-stage scheme used in the density-based explicit solver can efficiently remove local (high-frequency) errors as well. That is, the effect of the solution in one cell is communicated to adjacent cells relatively quickly. However, the scheme is less effective at reducing global (low-frequency) errors–errors that exist over a large number of control volumes. Thus, global corrections to the solution across a large number of control volumes occur slowly, over many iterations. This implies that performance of the multi-stage scheme will deteriorate as the number of control volumes increases.

Multigrid techniques allow global error to be addressed by using a sequence of successively coarser meshes. This method is based upon the principle that global (low-frequency) error existing on a fine mesh can be represented on a coarse mesh where it again becomes accessible as local (high-frequency) error: because there are fewer coarse cells overall, the global corrections can be communicated more quickly between adjacent cells. Since computations can be performed at an exponentially decaying expense in both CPU time and memory storage on coarser meshes, there is the potential for very efficient elimination of global error. The fine-grid relaxation scheme or "smoother", in this case either the point-implicit linear solvers (The Coupled and Scalar AMG Solvers (p. 1012)) or the explicit multi-stage scheme, is not required to be particularly effective at reducing global error and can be tuned for efficient reduction of local error.

24.7.1.2. The Basic Concept in Multigrid

Consider the set of discretized linear (or linearized) equations given by

$$A\varphi_e + b = 0 \tag{24.115}$$

where φ_e is the exact solution. Before the solution has converged there will be a defect d associated with the approximate solution φ:

$$A\varphi + b = d \tag{24.116}$$

We seek a correction ψ to φ such that the exact solution is given by

$$\varphi_e = \varphi + \psi \tag{24.117}$$

Substituting Equation 24.117 (p. 1004) into Equation 24.115 (p. 1004) gives

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$A(\phi+\psi)+b=0 \tag{24.118}$$

$$A\psi+(A\phi+b)=0 \tag{24.119}$$

Now using Equation 24.116 (p. 1004) and Equation 24.118 (p. 1005) we obtain

$$A\psi+d=0 \tag{24.120}$$

which is an equation for the correction in terms of the original fine level operator A and the defect d. Assuming the local (high-frequency) errors have been sufficiently damped by the relaxation scheme on the fine level, the correction ψ will be smooth and therefore more effectively solved on the next coarser level.

24.7.1.3. Restriction and Prolongation

Solving for corrections on the coarse level requires transferring the defect down from the fine level (restriction), computing corrections, and then transferring the corrections back up from the coarse level (prolongation). We can write the equations for coarse level corrections ψ^H as

$$A^H\psi^H+Rd=0 \tag{24.121}$$

where A^H is the coarse level operator and R the restriction operator responsible for transferring the fine level defect down to the coarse level. Solution of Equation 24.121 (p. 1005) is followed by an update of the fine level solution given by

$$\varphi^{new}=\varphi+P\psi^H \tag{24.122}$$

where P is the prolongation operator used to transfer the coarse level corrections up to the fine level.

24.7.1.4. Unstructured Multigrid

The primary difficulty with using multigrid on unstructured meshes is the creation and use of the coarse grid hierarchy. On a structured mesh, the coarse meshes can be formed simply by removing every other mesh line from the fine meshes and the prolongation and restriction operators are simple to formulate (for example, injection and bilinear interpolation).

The difficulties of applying multigrid on unstructured meshes are overcome in a separate fashion by each of the two multigrid methods used in Ansys Fluent. While the basic principles discussed so far and the cycling strategy described in Multigrid Cycles (p. 1005) are the same, the techniques for construction of restriction, prolongation, and coarse mesh operators are different, as discussed in Algebraic Multigrid (AMG) (p. 1009) and Full-Approximation Storage (FAS) Multigrid (p. 1014) for the AMG and FAS methods, respectively.

24.7.2. Multigrid Cycles

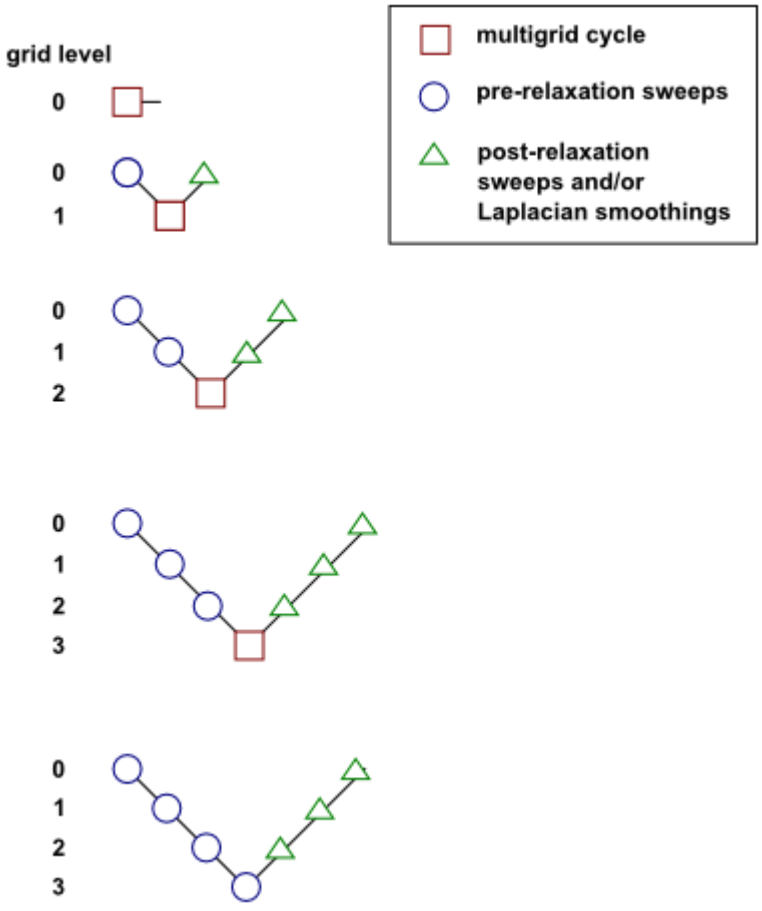
A multigrid cycle can be defined as a recursive procedure that is applied at each grid level as it moves through the grid hierarchy. Four types of multigrid cycles are available in Ansys Fluent: the V, W, F, and flexible ("flex") cycles. The V and W cycles are available in both AMG and FAS, while the F and flexible cycles are restricted to the AMG method only. (The W and flexible AMG cycles are not available for solving the coupled equation set due to the amount of computation required.)

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

24.7.2.1. The V and W Cycles

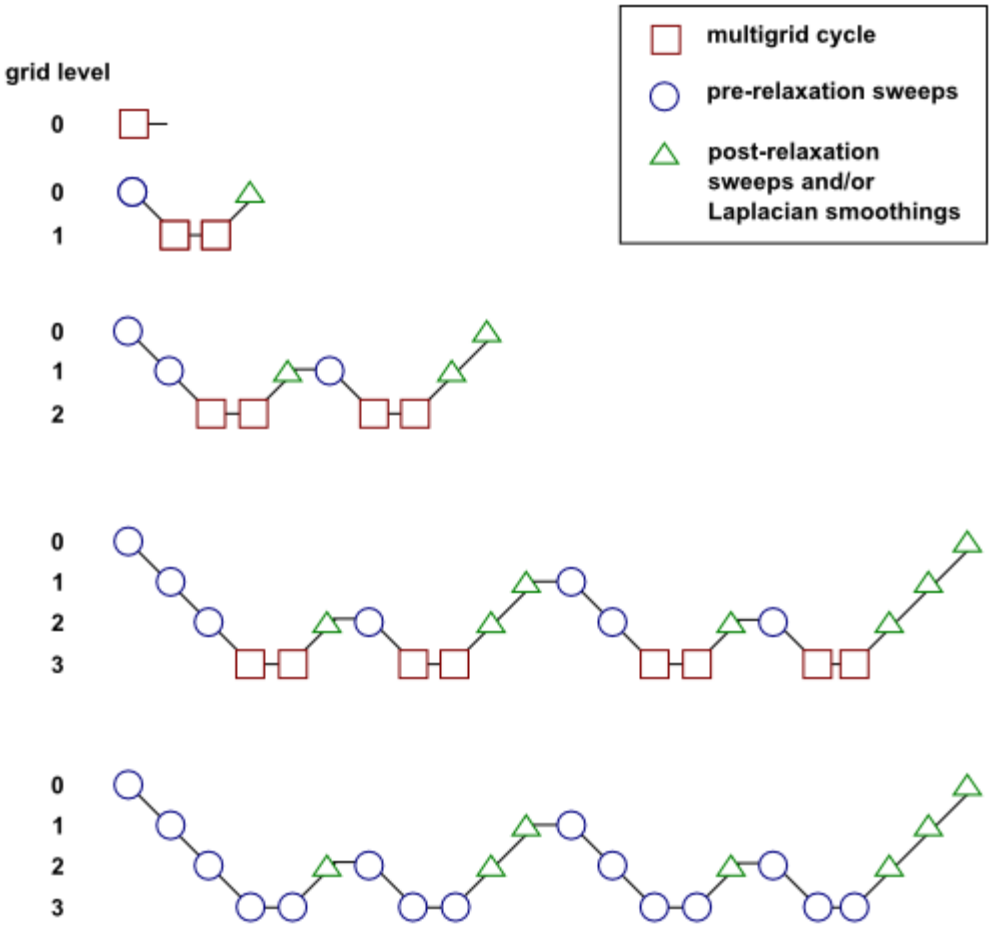
Figure 24.9: V-Cycle Multigrid (p. 1006) and Figure 24.10: W-Cycle Multigrid (p. 1007) show the V and W multigrid cycles (defined below). In each figure, the multigrid cycle is represented by a square, and then expanded recursively to show the individual steps that are performed within the cycle. The individual steps are represented by a circle, one or more squares, and a triangle, connected by lines: circle-square-triangle for a V cycle, or circle-square-square-triangle for a W cycle. The squares in this group expand again, into circle-square-triangle or circle-square-square-triangle, and so on. You may want to follow along in the figures as you read the steps below.

Figure 24.9: V-Cycle Multigrid



Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Figure 24.10: W-Cycle Multigrid



For the V and W cycles, the traversal of the hierarchy is governed by three parameters, β_1, β_2, and β_3, as follows:

1. First, iterations are performed on the current grid level to reduce the high-frequency components of the error (local error). For AMG, one iteration consists of one forward and one backward Gauss-Seidel sweep. For FAS, one iteration consists of one pass of the multi-stage scheme (described in Explicit Formulation (p. 996)).

   These iterations are referred to as pre-relaxation sweeps because they are performed before moving to the next coarser grid level. The number of pre-relaxation sweeps is specified by β_1.

   In Figure 24.9: V-Cycle Multigrid (p. 1006) and Figure 24.10: W-Cycle Multigrid (p. 1007) this step is represented by a circle and marks the start of a multigrid cycle. The high-wave-number components of error should be reduced until the remaining error is expressible on the next coarser mesh without significant aliasing.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

If this is the coarsest grid level, then the multigrid cycle on this level is complete. (In so the square representing the multigrid cycle on level 3 is equivalent to a circle, as shown in the final diagram in each figure.)

Important:

In the AMG method, the default value of β_1 is zero (that is, no pre-relaxation sweeps are performed).

2. Next, the problem is "restricted" to the next coarser grid level using Equation 24.121 (p. 1005).

   In Figure 24.9: V-Cycle Multigrid (p. 1006) and Figure 24.10: W-Cycle Multigrid (p. 1007), the restriction from a finer grid level to a coarser grid level is designated by a downward-sloping line.

3. The error on the coarse grid is reduced by performing a specified number (β_2) of multigrid cycles (represented in Figure 24.9: V-Cycle Multigrid (p. 1006) and Figure 24.10: W-Cycle Multigrid (p. 1007) as squares). Commonly, for fixed multigrid strategies β_2 is either 1 or 2, corresponding to V-cycle and W-cycle multigrid, respectively.

4. Next, the cumulative correction computed on the coarse grid is "interpolated" back to the fine grid using Equation 24.122 (p. 1005) and added to the fine grid solution. In the FAS method, the corrections are additionally smoothed during this step using the Laplacian smoothing operator discussed in Implicit Residual Smoothing (p. 996).

   In Figure 24.9: V-Cycle Multigrid (p. 1006) and Figure 24.10: W-Cycle Multigrid (p. 1007) the prolongation is represented by an upward-sloping line.

   The high-frequency error now present at the fine grid level is due to the prolongation procedure used to transfer the correction.

5. In the final step, iterations are performed on the fine grid to remove the high-frequency error introduced on the coarse grid by the multigrid cycles. These iterations are referred to as post-relaxation sweeps because they are performed after returning from the next coarser grid level. The number of post-relaxation sweeps is specified by β_3.

   In Figure 24.9: V-Cycle Multigrid (p. 1006) and Figure 24.10: W-Cycle Multigrid (p. 1007), this relaxation procedure is represented by a single triangle.

   For AMG, the default value of β_3 is 1.

   Important:

   Note, however, that if you are using AMG with V-cycle to solve an energy equation with a solid conduction model presented with anisotropic or very high conductivity coefficient, there is a possibility of divergence with a default post-relaxation sweep of 1. In such cases you should increase the post-relaxation sweep (for example, to 2) in the AMG section for better convergence, or change the cycle type to F-cycle or W-cycle, with an under-relaxation factor set to 1. This is especially effective when calculating pure heat conduction or conjugate heat transfer. Any instability observed

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

when using the F-cycle or W-cycle can be remedied by increasing the pre-relaxation sweep count to 1. Although the default value of 0 is optimal for most cases, increasing the pre-relaxation sweep value to 1 or 2 can improve convergence.

Since the default value for β_1 is 0 (that is, pre-relaxation sweeps are not performed), this procedure is roughly equivalent to using the solution from the coarse level as the initial guess for the solution at the fine level.

For FAS, the default value of β_3 is zero (that is, post-relaxation sweeps are not performed); post-relaxation sweeps are never performed at the end of the cycle for the finest grid level, regardless of the value of β_3. This is because for FAS, post-relaxation sweeps at the fine level are equivalent to pre-relaxation sweeps during the next cycle.

24.7.3. Algebraic Multigrid (AMG)

This algorithm is referred to as an "algebraic" multigrid scheme because, as we shall see, the coarse level equations are generated without the use of any geometry or re-discretization on the coarse levels; a feature that makes AMG particularly attractive for use on unstructured meshes. The advantage being that no coarse meshes have to be constructed or stored, and no fluxes or source terms need to be evaluated on the coarse levels. This approach is in contrast with FAS (sometimes called "geometric") multigrid in which a hierarchy of meshes is required and the discretized equations are evaluated on every level. In theory, the advantage of FAS over AMG is that the former should perform better for non-linear problems since non-linearities in the system are carried down to the coarse levels through the re-discretization; when using AMG, once the system is linearized, non-linearities are not "felt" by the solver until the fine level operator is next updated.

24.7.3.1. AMG Restriction and Prolongation Operators

The restriction and prolongation operators used here are based on the additive correction (AC) strategy described for structured meshes by Hutchinson and Raithby [264] (p. 1068). Inter-level transfer is accomplished by piecewise constant interpolation and prolongation. The defect in any coarse level cell is given by the sum of those from the fine level cells it contains, while fine level corrections are obtained by injection of coarse level values. In this manner the prolongation operator is given by the transpose of the restriction operator

$$P=R^T \tag{24.123}$$

The restriction operator is defined by a coarsening or "grouping" of fine level cells into coarse level ones. In this process each fine level cell is grouped with one or more of its "strongest" neighbors, with a preference given to currently ungrouped neighbors. The algorithm attempts to collect cells into groups of fixed size, typically two or four, but any number can be specified.

By default, the “strongest” neighbor is evaluated as the neighbor j of the current cell i for which the coefficient A_{ij} is largest. For sets of coupled equations A_{ij} is a block matrix and the measure of its magnitude is simply taken to be the magnitude of its first element. In addition, the set of coupled equations for a given cell are treated together and not divided amongst different coarse cells. This results in the same coarsening for each equation in the system.

You have the option of enabling a conservative coarsening scheme that tunes the multigrid coarsening based on coefficient strengths and partitioning, as well as an aggressive coarsening

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

scheme that is optimized for higher coarsening rates. Alternatively, you can enable Laplace coarsening so that Ansys Fluent uses Laplace coefficients as the strength factors in determining the coarsening. For details, refer to Coarsening Parameters in the *Fluent User's Guide*.

24.7.3.2. AMG Coarse Level Operator

The coarse level operator A^H is constructed using a Galerkin approach. Here we require that the defect associated with the corrected fine level solution must vanish when transferred back to the coarse level. Therefore we may write

$$Rd^{new}=0 \tag{24.124}$$

Upon substituting Equation 24.116 (p. 1004) and Equation 24.122 (p. 1005) for d^{new} and φ^{new} we have

$$\begin{aligned} R\left[A\varphi^{new}+b\right]&=0 \\ R\left[A\left(\varphi+P\psi^H\right)+b\right]&=0 \end{aligned} \tag{24.125}$$

Now rearranging and using Equation 24.116 (p. 1004) once again gives

$$\begin{aligned} RAP\psi^H+R\left(A\varphi+b\right)&=0 \\ RAP\psi^H+Rd&=0 \end{aligned} \tag{24.126}$$

Comparison of Equation 24.126 (p. 1010) with Equation 24.121 (p. 1005) leads to the following expression for the coarse level operator:

$$A^H=RAP \tag{24.127}$$

The construction of coarse level operators therefore reduces to a summation of diagonal and corresponding off-diagonal blocks for all fine level cells within a group to form the diagonal block of that group's coarse cell.

24.7.3.3. The F Cycle

The multigrid F cycle is essentially a combination of the V and W cycles described in The V and W Cycles (p. 1006).

Recall that the multigrid cycle is a recursive procedure. The procedure is expanded to the next coarsest grid level by performing a single multigrid cycle on the current level. Referring to Figure 24.9: V-Cycle Multigrid (p. 1006) and Figure 24.10: W-Cycle Multigrid (p. 1007), this means replacing the square on the current level (representing a single cycle) with the procedure shown for the 0-1 level cycle (the second diagram in each figure). We see that a V cycle consists of:

$$\text{pre sweep}\rightarrow\text{restrict}\rightarrow\text{V cycle}\rightarrow\text{prolongate}\rightarrow\text{post sweep} \tag{24.128}$$

and a W cycle:

$$\text{pre sweep}\rightarrow\text{restrict}\rightarrow\text{W cycle}\rightarrow\text{W cycle}\rightarrow\text{prolongate}\rightarrow\text{post sweep} \tag{24.129}$$

An F cycle is formed by a W cycle followed by a V cycle:

$$\text{pre sweep}\rightarrow\text{restrict}\rightarrow\text{W cycle}\rightarrow\text{V cycle}\rightarrow\text{prolongate}\rightarrow\text{post sweep} \tag{24.130}$$

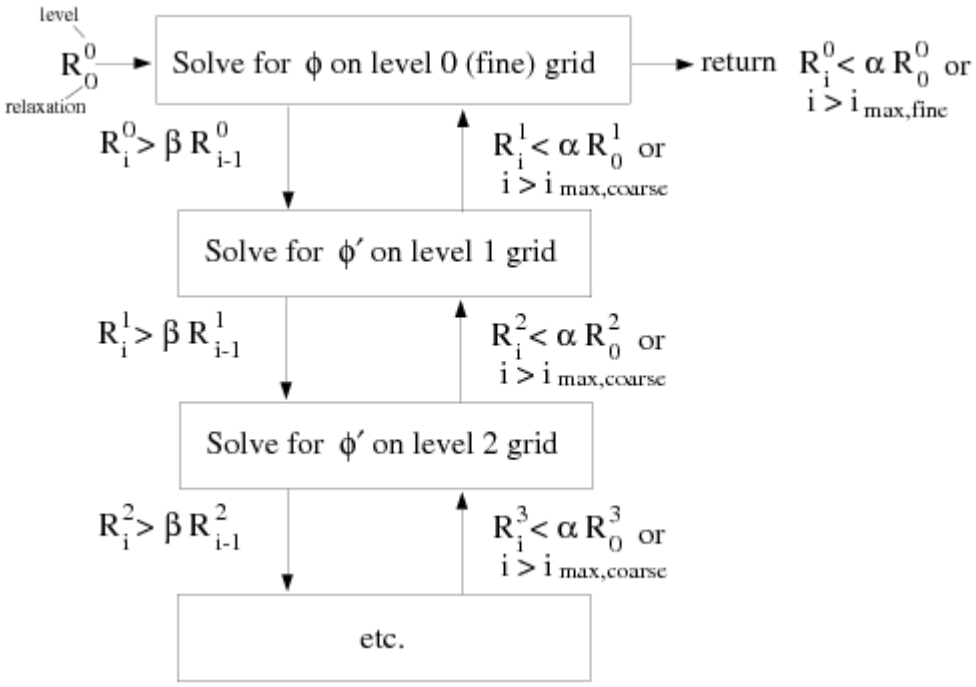
As expected, the F cycle requires more computation than the V cycle, but less than the W cycle. However, its convergence properties turn out to be better than the V cycle and roughly equivalent to the W cycle. The F cycle is the default AMG cycle type for the coupled equation set and for the scalar energy equation.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

24.7.3.4. The Flexible Cycle

For the flexible cycle, the calculation and use of coarse grid corrections is controlled in the multigrid procedure by the logic illustrated in Figure 24.11: Logic Controlling the Flex Multigrid Cycle (p. 1011). This logic ensures that coarser grid calculations are invoked when the rate of residual reduction on the current grid level is too slow. In addition, the multigrid controls dictate when the iterative solution of the correction on the current coarse grid level is sufficiently converged and should therefore be applied to the solution on the next finer grid. These two decisions are controlled by the parameters α and β shown in Figure 24.11: Logic Controlling the Flex Multigrid Cycle (p. 1011), as described in detail below. Note that the logic of the multigrid procedure is such that grid levels may be visited repeatedly during a single global iteration on an equation. For a set of 4 multigrid levels, referred to as 0, 1, 2, and 3, the flex-cycle multigrid procedure for solving a given transport equation might consist of visiting grid levels as 0-1-2-3-2-3-2-1-0-1-2-1-0, for example.

Figure 24.11: Logic Controlling the Flex Multigrid Cycle



The main difference between the flexible cycle and the V and W cycles is that the satisfaction of the residual reduction tolerance and termination criterion determine when and how often each level is visited in the flexible cycle, whereas in the V and W cycles the traversal pattern is explicitly defined.

24.7.3.4.1. The Residual Reduction Rate Criteria

The multigrid procedure invokes calculations on the next coarser grid level when the error reduction rate on the current level is insufficient, as defined by

$$R_i > \beta R_{i-1} \tag{24.131}$$

Here R_i is the absolute sum of residuals (defect) computed on the current grid level after the ith relaxation on this level. The above equation states that if the residual present in the iterative solution after i relaxations is greater than some fraction, β (between 0 and 1), of the residual present after the $(i-1)$th relaxation, the next coarser grid level should be visited. Thus β is referred

to as the residual reduction tolerance, and determines when to "give up" on the iterative solution at the current grid level and move to solving the correction equations on the next coarser grid. The value of β controls the frequency with which coarser grid levels are visited. The default value is 0.7. A larger value will result in less frequent visits, and a smaller value will result in more frequent visits.

24.7.3.4.2. The Termination Criteria

Provided that the residual reduction rate is sufficiently rapid, the correction equations will be converged on the current grid level and the result applied to the solution field on the next finer grid level.

The correction equations on the current grid level are considered sufficiently converged when the error in the correction solution is reduced to some fraction, α (between 0 and 1), of the original error on this grid level:

$$R_i < \alpha R_0 \tag{24.132}$$

Here, R_i is the residual on the current grid level after the ith iteration on this level, and R_0 is the residual that was initially obtained on this grid level at the current global iteration. The parameter α, referred to as the termination criterion, has a default value of 0.1. Note that the above equation is also used to terminate calculations on the lowest (finest) grid level during the multigrid procedure. Thus, relaxations are continued on each grid level (including the finest grid level) until the criterion of this equation is obeyed (or until a maximum number of relaxations has been completed, in the case that the specified criterion is never achieved).

24.7.3.5. The Coupled and Scalar AMG Solvers

The scalar AMG solver is used for the solution of linear systems obtained from the discretization of the individual transport equations.

$$a_{ij} x_j = b_i \tag{24.133}$$

where the above equation contains scalar variables.

The coupled AMG solver is used to solve linear transport equations using implicit discretization from coupled systems such as flow variables for the density-based solver, pressure-velocity variables for the coupled pressure-based schemes and inter-phase coupled individual equations for Eulerian multiphase flows.

$$[A]_{ij} \vec{X}_j = \vec{B}_i \tag{24.134}$$

where the influence of a cell i on a cell j has the form

$$A_{ij} = \begin{bmatrix} a_{ij}^{11} & a_{ij}^{12} & \dots & a_{ij}^{1N} \\ a_{ij}^{21} & . & & . \\ \vdots & & . & . \\ a_{ij}^{N1} & & & a_{ij}^{NN} \end{bmatrix} \tag{24.135}$$

and the unknown and source vectors have the form

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$\vec{X}_j = \begin{bmatrix} x_j^1 \\ \vdots \\ x_j^N \end{bmatrix} \tag{24.136}$$

$$\vec{B}_i = \begin{bmatrix} b_i^1 \\ \vdots \\ b_i^N \end{bmatrix} \tag{24.137}$$

The above resultant system of equations is solved in Ansys Fluent using either the Gauss-Seidel smoother or the Incomplete Lower Upper decomposition (ILU) smoother. If a scalar system of equations is to be solved then the point-method (Gauss-Seidel or ILU) smoother is used, while for a coupled system of equations the block-method (Gauss-Seidel or ILU) smoother is used.

24.7.3.5.1. Gauss-Seidel

The Gauss-Seidel method is a technique for solving a linear system of equations one at a time and in sequence. It uses the previously computed results as soon as they become available. It performs two sweeps on the unknowns in forward and backward directions. Both point or block method Gauss-Seidel smoothers are available in Ansys Fluent to solve for either the scalar AMG system of equations or the coupled AMG system of equations.

The Gauss-Seidel procedure can be illustrated using the scalar system, Equation 24.133 (p. 1012). The forward sweep can be written as:

$$x_i^{k+1/2} = \left(b_i - \sum_{j<i} a_{ij} x_j^{k+1/2} - \sum_{j>i} a_{ij} x_j^k \right) / \, a_{ii} \quad (i = 1, \ldots, N) \tag{24.138}$$

where N is the number of unknowns. The forward sweep is followed by a backward sweep that can be written as:

$$x_i^{k+1} = \left(b_i - \sum_{j<i} a_{ij} x_j^{k+1/2} - \sum_{j>i} a_{ij} x_j^{k+1} \right) / \, a_{ii} \tag{24.139}$$

Following from Equation 24.138 (p. 1013) and Equation 24.139 (p. 1013), symmetric Gauss-Seidel can be expressed in matrix form as a two-step recursive solution of the system

$$(D_A + L_A) D_A^{-1} (D_A + U_A)(x^{k+1} - x^k) = b - Ax^k \tag{24.140}$$

where D_A, L_A, and U_A represent diagonal, lower tridiagonal, and upper tridiagonal parts of matrix A, respectively.

Symmetric Gauss-Seidel has a somewhat limited rate of smoothing of residuals between levels of AMG, unless the coarsening factor is set to 2.

24.7.3.5.2. Incomplete Lower Upper (ILU)

A more effective AMG smoother is based on the ILU decomposition technique. In general, any iterating method can be represented as

$$M\left(x^{k+1} - x^k\right) = b - Ax^k \tag{24.141}$$

where matrix M is some approximation of the original matrix A from

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$Ax=b \tag{24.142}$$

M should be close to A and the calculation of M^{-1} should have a low operation count. We consider M as an incomplete lower upper factorization of the matrix A such that

$$M=LU=\left(D+L_A\right)D^{-1}\left(D+U_A\right) \tag{24.143}$$

where L_A and U_A are the lower tridiagonal and upper tridiagonal parts of matrix A. The diagonal matrix D is calculated in a special way to satisfy the following condition for diagonal D_M of matrix M:

$$D_M=D_A \tag{24.144}$$

In this case, the ith element of the diagonal of D will be calculated using

$$d_{ii}=a_{ii}-\sum_{j<1}\left(\frac{a_{ij}a_{ji}}{d_{jj}}\right) \tag{24.145}$$

The calculation of the new solution x^{k+1} is then performed in two symmetric recursive sweeps, similar to Gauss-Seidel sweeps. Diagonal elements d_{ii} of the ILU decomposition are calculated during the construction of levels and stored in the memory. ILU smoother is slightly more expensive compared to Gauss-Seidel, but has better smoothing properties, especially for block-coupled systems solved by coupled AMG. In this case, coarsening of levels can be more aggressive using coarsening factors between 8 and 12 for 3D problems compared to 2 for Gauss-Seidel.

Important:

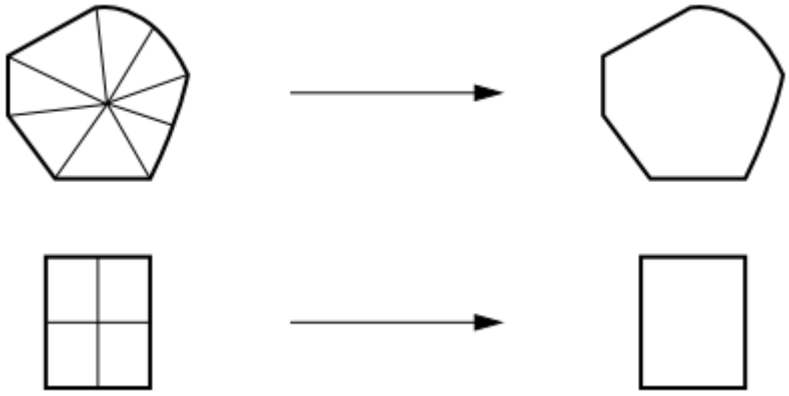
When solving the coupled systems, shorter solution times and more robust performance can be obtained by using the default ILU smoother, rather than the Gauss-Seidel smoother, which is the default for scalar systems. ILU is recommended whenever the coupled AMG solver is used. It is used by default for the coupled equations and, when a pseudo time method is selected, for the scalar equations.

24.7.4. Full-Approximation Storage (FAS) Multigrid

Ansys Fluent's approach to forming the multigrid grid hierarchy for FAS is simply to coalesce groups of cells on the finer grid to form coarse grid cells. Coarse grid cells are created by agglomerating the cells surrounding a node, as shown in Figure 24.12: Node Agglomeration to Form Coarse Grid Cells (p. 1015). Depending on the grid topology, this can result in cells with irregular shapes and variable numbers of faces. The grid levels are, however, simple to construct and are embedded, resulting in simple prolongation and relaxation operators.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Figure 24.12: Node Agglomeration to Form Coarse Grid Cells



It is interesting to note that although the coarse grid cells look very irregular, the discretization cannot "see" the jaggedness in the cell faces. The discretization uses only the area projections of the cell faces and therefore each group of "jagged" cell faces separating two irregularly-shaped cells is equivalent to a single straight line (in 2D) connecting the endpoints of the jagged segment. (In 3D, the area projections form an irregular, but continuous, geometrical shape.) This optimization decreases the memory requirement and the computation time.

24.7.4.1. FAS Restriction and Prolongation Operators

FAS requires restriction of both the fine grid solution φ and its residual (defect) d. The restriction operator R used to transfer the solution to the next coarser grid level is formed using a full-approximation scheme [75] (p. 1057). That is, the solution for a coarse cell is obtained by taking the volume average of the solution values in the embedded fine grid cells. Residuals for the coarse grid cell are obtained by summing the residuals in the embedded fine grid cells.

The prolongation operator P used to transfer corrections up to the fine level is constructed to simply set the fine grid correction to the associated coarse grid value.

The coarse grid corrections ψ^H, which are brought up from the coarse level and applied to the fine level solution, are computed from the difference between the solution calculated on the coarse level φ^H and the initial solution restricted down to the coarse level $R\varphi$. Thus correction of the fine level solution becomes

$$\varphi^{new} = \varphi + P\left(\varphi^H - R\varphi\right) \tag{24.146}$$

24.7.4.2. FAS Coarse Level Operator

The FAS coarse grid operator A^H is simply that which results from a re-discretization of the governing equations on the coarse level mesh. Since the discretized equations presented in Discretization (p. 962) and Density-Based Solver (p. 990) place no restrictions on the number of faces that make up a cell, there is no problem in performing this re-discretization on the coarse grids composed of irregularly shaped cells.

There is some loss of accuracy when the finite-volume scheme is used on the irregular coarse grid cells, but the accuracy of the multigrid solution is determined solely by the finest grid and is therefore not affected by the coarse grid discretization.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

In order to preserve accuracy of the fine grid solution, the coarse level equations are modified to include source terms [275] (p. 1068) which insure that corrections computed on the coarse grid φ^H will be zero if the residuals on the fine grid d^h are zero as well. Thus, the coarse grid equations are formulated as

$$A^H\varphi^H + d^H = d^H\left(R\varphi\right) - Rd^h \tag{24.147}$$

Here d^H is the coarse grid residual computed from the current coarse grid solution φ^H, and $d^H\left(R\varphi\right)$ is the coarse grid residual computed from the restricted fine level solution $R\varphi$. Initially, these two terms will be the same (because initially we have $\varphi^H = R\varphi$) and cancel from the equation, leaving

$$A^H\varphi^H = -Rd^h \tag{24.148}$$

So there will be no coarse level correction when the fine grid residual d^h is zero.

24.8. Hybrid Initialization

Hybrid Initialization is a collection of recipes and boundary interpolation methods. It solves the Laplace equation to produce a velocity field that conforms to complex domain geometries, and a pressure field which smoothly connects high and low pressure values in the computational domain. All other variables (that is temperature, turbulence, VOF, species, and so on) will be patched based on domain averaged values or a predetermined recipe. These recipes are detailed below.

- **Velocity Field:** Laplace's equation is solved with appropriate boundary conditions to produce the velocity field in the domain.

  $$\nabla^2\varphi = 0 \tag{24.149}$$

  where φ is the velocity potential. The velocity components are given by the gradient potential:

  $$\vec{V} = \nabla\varphi \tag{24.150}$$

  Velocity potential is expressed as follows for the various boundary conditions:

  – **Wall Boundaries:** The velocity normal to the wall is zero.

  $$\left.\frac{\partial\varphi}{\partial n}\right|_{wall} = 0 \tag{24.151}$$

  – **Inlet Boundaries:** The velocity normal to the inlet boundaries are computed from the user-specified boundary values.

  $$\left.\frac{\partial\varphi}{\partial n}\right|_{inlet} = V_\perp \tag{24.152}$$

  – **Far Field Boundaries:** At the far-field boundaries the velocity normal to the boundaries is computed from the user-specified free stream conditions.

  $$\left.\frac{\partial\varphi}{\partial n}\right|_{inlet} = V_\perp \tag{24.153}$$

  Far away from the body the flow approaches the free stream conditions:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$\left.\frac{\partial \varphi}{\partial x}\right|_{\infty} = u_{\infty} \tag{24.154}$$

$$\left.\frac{\partial \varphi}{\partial y}\right|_{\infty} = v_{\infty} \tag{24.155}$$

$$\left.\frac{\partial \varphi}{\partial z}\right|_{\infty} = w_{\infty} \tag{24.156}$$

  - **Outlet Boundaries:** They are specified as zero potential.

$$\varphi = 0 \tag{24.157}$$

  A uniform initial velocity magnitude with the direction taken from the solution of the velocity potential solution can be specified by selecting the **Maintain Constant Velocity Magnitude** option under the **General Setting** tab.

- **Pressure Field:** An additional Laplace equation is solved with the appropriate boundary condition to produce the smooth pressure field in the domain, only if the pressure information is available in at least one inlet and one outlet in the domain. Otherwise, the pressure field will be initialized with the averaged constant value from all the boundaries.

$$\nabla^2 P = 0 \tag{24.158}$$

  where P is expressed as follows for the various boundary conditions:

  - Pressure inlet boundaries: by default, first an approximate check is made to determine whether the boundary flow is supersonic or subsonic, where a large difference between the specified **Gauge Total Pressure** and **Supersonic/Initial Gauge Pressure** corresponds to supersonic flow, and a moderate / small (or zero) pressure difference corresponds to subsonic flow. When a boundary is determined to be supersonic, then P uses the value of the specified **Supersonic/Initial Gauge Pressure**; when the boundary is determined to be subsonic, then P uses the value of the specified **Gauge Total Pressure**. If you enabled the **Use Specified Pressure on Inlets** option in the **General Settings** tab of the **Hybrid Initialization** dialog box, then P always uses the specified **Supersonic/Initial Gauge Pressure**.

  - Pressure outlet and Pressure Far-Field boundaries: P is computed as the specified **Gauge Pressure** at this boundary.

  - Velocity/Mass-flow inlet boundaries: P uses the value of the specified **Supersonic/Initial Gauge Pressure** only if you have selected the **Use specified pressure on the inlet** option under the **General Settings** tab.

  - Wall Boundaries: the normal gradient of P is set to zero.

$$\left.\frac{\partial p}{\partial n}\right|_{wall} = 0 \tag{24.159}$$

- **Temperature Field:** Temperature will be initialized with a constant value (domain averaged).

- **Turbulent Parameters:** By default, turbulent parameters are initialized with constant values (domain averaged). However, if you want to initialize the flow with variable turbulence parameters, then you will need to select the appropriate option in the **Hybrid Initialization** dialog box, described in Steps in Using Hybrid Initialization.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

- **Species Fractions:** By default, secondary species mass/mole fractions are initialized with a 0.0 value. However, if you want to specify the initialization values, then you will need to select the appropriate option in the **Hybrid Initialization** dialog box, described in Steps in Using Hybrid Initialization.
- **Volume Fractions:** For a two-phase case, the secondary VOF phase will be initialized with the minimum VOF from all inlets. If the case has more than two phases, then the secondary phases will be initialized with a zero VOF.

For information about using hybrid initialization in Ansys Fluent, see Hybrid Initialization in the User's Guide.

24.9. Full Multigrid (FMG) Initialization

For many complex flow problems such as those found in rotating machinery, or flows in expanding or spiral ducts, flow convergence can be accelerated if a better initial solution is used at the start of the calculation. The Full Multigrid initialization (FMG initialization) can provide this initial and approximate solution at a minimum cost to the overall computational expense.

For more information about using FMG initialization in Ansys Fluent, see Full Multigrid (FMG) Initialization in the User's Guide.

There are additional options that can be used with the density-based solver. For details, see Additional FMG Initialization Options with the Density-Based Solver in the *Fluent User's Guide*.

For more information about FMG initialization, see the following sections:

24.9.1. Overview of FMG Initialization

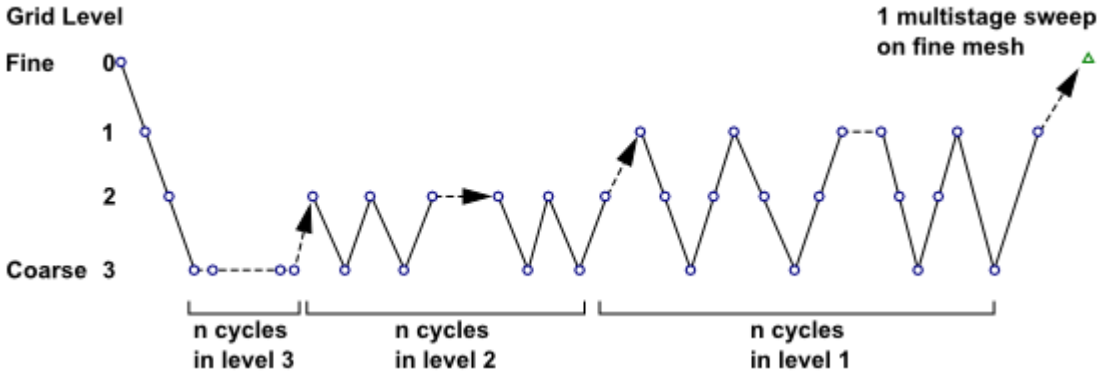
24.9.2. Limitations of FMG Initialization

24.9.1. Overview of FMG Initialization

FMG initialization utilizes the Ansys Fluent FAS Multigrid technology (see Full-Approximation Storage (FAS) Multigrid (p. 1014)) to obtain the initial solution. Starting from a uniform solution (after performing standard or hybrid initialization or reading data), the FMG initialization procedure constructs the desirable number of geometric grid levels using the procedure outlined in Full-Approximation Storage (FAS) Multigrid (p. 1014). To begin the process, the initial solution is restricted all the way down to the coarsest level. The FAS multigrid cycle is then applied until a given order of residual reduction is obtained or the maximum number of cycles is reached. The solution is then interpolated one grid level up and the FAS multigrid cycle is applied again between the current level all the way down to the coarsest level. This process will repeat until the finest level is reached. The FMG initialization iteration is illustrated in Figure 24.13: The FMG Initialization (p. 1019).

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Figure 24.13: The FMG Initialization



Since FMG initialization does most of the work on coarse levels, this initialization procedure is computationally inexpensive and, for large problems, a good initial solution can be obtained in a fraction of the time spent to converge on a final solution. Note that FMG initialization can be used with the pressure-based and density-based solvers.

When FMG initialization is started, the algorithm will perform the following steps:

1. Records the current solver selection and all current solver parameters.
2. Switches from the selected solver to the density-based explicit formulation.
3. Performs one FMG iteration using the FMG parameters given in the text command interface (see below).
4. Switches back to the initially selected solver and resets all solver parameters back to the original solver settings.

In the FMG iteration, the Euler equations for inviscid flow are solved using first order-discretization to obtain the approximate solution. If species are present, then the FMG initialization will solve the species equations. While turbulence equations or any other transport scalars are not solved during the FMG initialization cycles, their values will be updated once after the final FMG sweep on the fine mesh using the latest velocity and pressure fields

24.9.2. Limitations of FMG Initialization

- FMG initialization is not available for unsteady flows.

> **Important:**
>
> If an initial solution is needed for an unsteady calculation, then you must first switch to the steady-state solver, perform FMG initialization and other necessary iterations to get an initial solution, and then perform unsteady calculations.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

- FMG will not solve turbulence or other transport equations field variables.

  Important:

  If you are solving for turbulent flow, then you can still use FMG initialization, however, while turbulence equations or any other transport scalars are not solved during the FMG initialization cycles, their values will be updated once after the final FMG sweep on the fine mesh using the latest velocity and pressure fields.

- FMG can not be used with multiphase flow.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Chapter 25: Adapting the Mesh

The solution-adaptive mesh refinement feature of Ansys Fluent allows you to refine and/or coarsen your mesh based on geometric and numerical solution data. In addition, Ansys Fluent provides tools for creating and viewing adaption fields customized to particular applications. For information about using mesh adaption in Ansys Fluent, see Adapting the Mesh in the User's Guide. Theoretical information about the adaption process is described in detail in the following sections.

25.1. Adaption Process

25.2. Geometry-Based Adaption

25.1. Adaption Process

The adaption process is separated into two distinct tasks.

1. The individual cells are marked for refinement or coarsening based on the adaption function, which is created from geometric and/or solution data.

2. The cell is refined or considered for coarsening based on these adaption marks.

   The primary advantages of this modularized approach are the abilities to create sophisticated adaption functions and to experiment with various adaption functions without modifying the existing mesh.

Important:

Write a case and data file before starting the adaption process. If you generate an undesirable mesh, you can restart the process with the saved files.

Two methods of adaption are available in Ansys Fluent:

- hanging node adaption

  This method refines various cell types based on predefined templates (see Hanging Node Adaption (p. 1022)). It does not refine polyhedral cells, and requires additional memory to maintain the hierarchy of mesh refinement levels.

- polyhedral unstructured mesh adaption (PUMA)

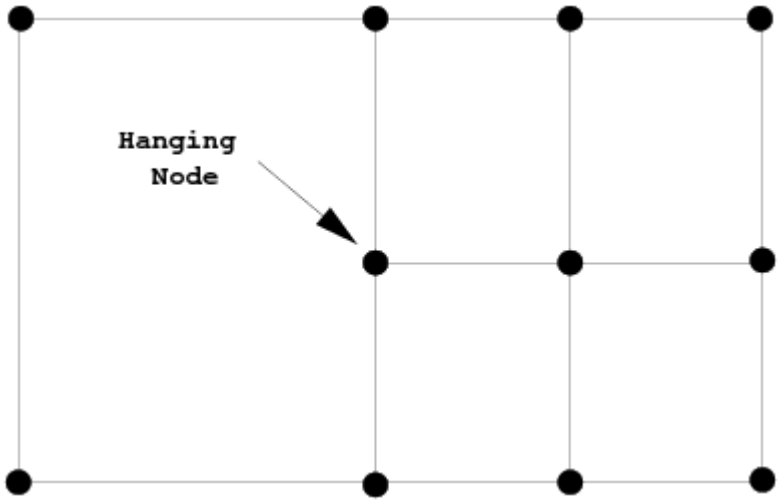
  This method treats all cell types as polyhedra, and applies a single refinement methodology (see Polyhedral Unstructured Mesh Adaption (p. 1024)). It does not create hanging nodes and requires less memory than hanging node adaption. It is only available for 3D meshes (for which it is the default), and is available in a version that is suitable for 2.5D meshes (that is, meshes that are one cell thick).

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

25.1.1. Hanging Node Adaption

Hanging node adaption is used for 2D meshes, and can be selected for 3D meshes. The meshes produced by this method are characterized by nodes on edges and faces that are not vertices of all the cells sharing those edges or faces, as shown in Figure 25.1: Example of a Hanging Node (p. 1022).

Figure 25.1: Example of a Hanging Node



The hanging node adaption method uses a variety of predefined templates to recursively refine cells in the mesh. As implemented in Ansys Fluent it can refine triangles, quadrilaterals, tetrahedra, hexahedra, wedges, and pyramids, but not polyhedra. It requires additional memory to maintain the hierarchy of mesh refinement levels, which is necessary for the mesh coarsening procedure that recovers the original mesh.

The cells marked for refinement are divided as described here:

- A triangle is split into 4 triangles.
- A quadrilateral is split into 4 quadrilaterals.
- A tetrahedron is split into eight tetrahedra. The subdivision consists of trimming each corner of the tetrahedron, and subdividing the enclosed octahedron by introducing the shortest diagonal.
- A hexahedron is split into 8 hexahedra.
- A wedge is split into 8 wedges.
- A pyramid is split into 6 pyramids and 4 tetrahedra.

Figure 25.2: Hanging Node Adaption of 2D Cell Types (p. 1023) and Figure 25.3: Hanging Node Adaption of 3D Cell Types (p. 1023) illustrate the division of the supported cell shapes.

To maintain accuracy, neighboring cells are not allowed to differ by more than one level of refinement. This prevents the adaption from producing excessive cell volume variations (reducing truncation error) and ensures that the positions of the parent (original) and child (refined) cell centroids are similar (reducing errors in the flux evaluations).

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Figure 25.2: Hanging Node Adaption of 2D Cell Types

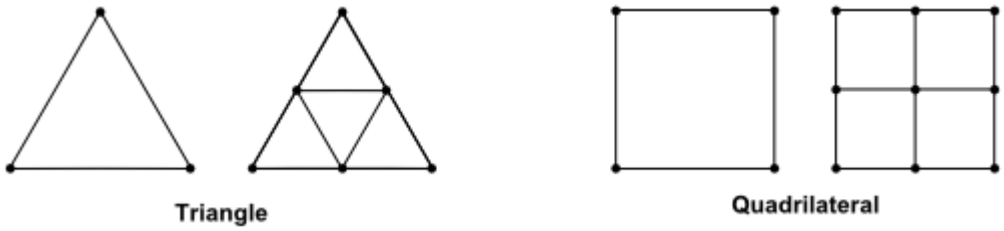
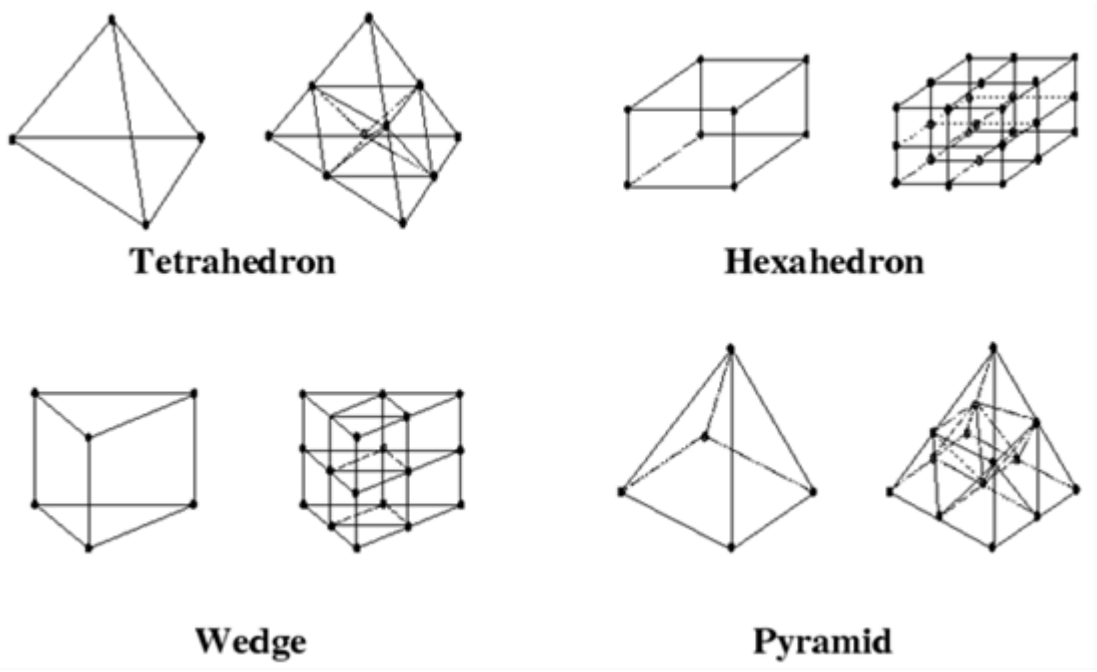


Figure 25.3: Hanging Node Adaption of 3D Cell Types



The mesh can be coarsened after it has been refined, by reintroducing inactive parent cells (uniting the child cells to reclaim the previously subdivided parent cell). An inactive parent cell is reactivated if all its children are marked for coarsening. You can eventually reclaim the original mesh with repeated application of the coarsening. Through this process, it is not possible to coarsen the mesh further than the original mesh.

The hanging node method has the following limitations:

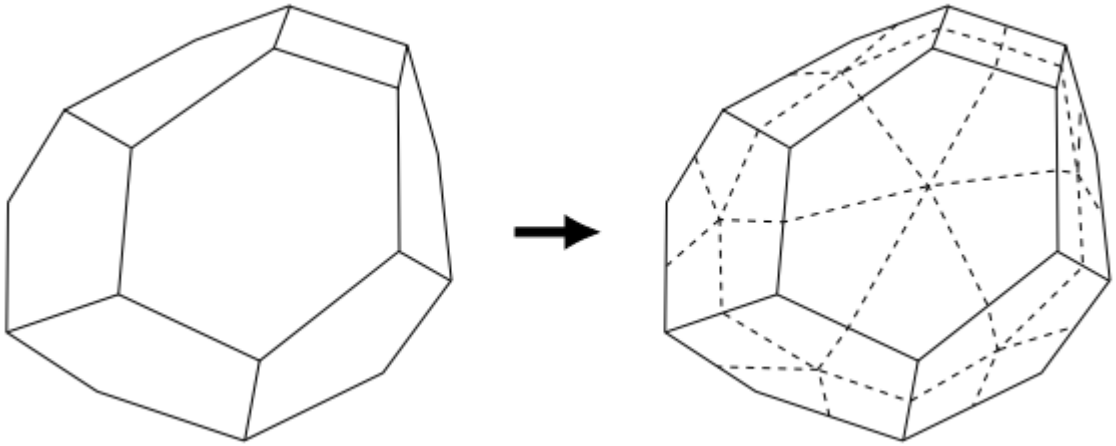
- Polyhedra will not be refined when this method is applied.
- The hanging nodes in the adapted mesh are incompatible with dynamic layering and remeshing methods.
- Face and/or cell zone separation should be done prior to hanging node adaption, otherwise you would need to convert cells with hanging nodes to polyhedra.
- Legacy anisotropic or prismatic refinement is not supported for meshes that have undergone hanging node adaption.
- Fusing of face zones is not supported for hanging-node adapted meshes.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

25.1.2. Polyhedral Unstructured Mesh Adaption

The polyhedral unstructured mesh adaption (PUMA) method is only available for 3D meshes, and is selected by default. This method generalizes the adaption technique to meshes containing arbitrary polyhedra. Since it does not rely on templates for refinement, it is not limited to particular cell types and can refine any 3D type (polyhedra, tetrahedra, hexahedra, and so on). It also consumes less memory during mesh refinement when compared to the hanging node method.

Figure 25.4: PUMA Refinement of a Polyhedral Cell



A mesh can be coarsened after it has been refined, such that the child cells are agglomerated to reclaim the previously subdivided parent cell. A parent cell can only be reclaimed if all its children are marked for coarsening. You can eventually reclaim the original mesh with repeated application of the coarsening. Through this process, it is not possible to coarsen the mesh further than the original mesh.

The PUMA method has the following limitations:

- It is only available for 3D cases.
- Non-convex polyhedra are not guaranteed to be refined; they will be skipped if it is not possible to yield child cells that have a positive volume and an acceptable orthogonal quality.
- The adapted mesh is very likely to include polyhedral cells, which are incompatible with remeshing methods.
- The writing of adaption hierarchy and refinement history to file is not supported if you choose to write your case file in the legacy format (as described in Reading and Writing Files in the Legacy Format in the *Fluent User's Guide*). For this format, the hierarchy information is lost when a mesh adapted using this method is written to a case file, and so there are limitations if the mesh is read into a new session of Ansys Fluent: coarsening is not possible, and further refinement would produce different results than if it was performed in the original session.
- The PUMA method cannot be applied to meshes that have a mesh hierarchy (for example, hexcore-type meshes). You may be able to remove the mesh hierarchy by converting the mesh to polyhedra using the `mesh/polyhedra/convert-hanging-nodes` text command, or (for cases that have undergone hanging node adaption) by coarsening back to the original mesh.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

- Legacy anisotropic or prismatic refinement is not supported for meshes that have undergone adaption with the PUMA method.
- Fusing of face zones is only supported for serial cases.

25.2. Geometry-Based Adaption

The purpose of adaption is to produce a mesh that is fine enough and adequately represents all important features of the geometry. However, when you have a coarse mesh of a geometry that has curved profiles and sharp corners, the adapted mesh may not recover the curved profiles and corners at the perimeter of the geometry. In such cases, you can use geometry-based adaption to reconstruct the geometry (or to recover the finer details of the geometry), while you are also performing the adaption process.

For more information, see the following section:

25.2.1. Geometry-Based Adaption Approach

25.2.1. Geometry-Based Adaption Approach

Geometry-based adaption works on the principle of geometry reconstruction. In this approach, the cell count of the mesh is increased by creating the new nodes in the domain in between the existing nodes of the mesh. The newly created nodes are projected in such a way that the resulting mesh is finer and has a shape that is closer to the original geometry.

The following sections explain how nodes are projected and the parameters that control the node propagation.

25.2.1.1. Node Projection

Consider a coarse mesh created for a circular geometry. A section of the mesh close to the circular edge is shown in Figure 25.5: Mesh Before Adaption (p. 1026). The edge is not smooth and has sharp corners, such that its shape does not closely resemble that of the original geometry. Using boundary adaption along with the geometry reconstruction option will result in a mesh with smoother edges, as shown in Figure 25.6: Projection of Nodes (p. 1026).

In Figure 25.6: Projection of Nodes (p. 1026), the dotted lines represent the original edge of the mesh. The boundary adaption process creates new nodes in between the original nodes. These nodes are projected towards the edge of the geometry, and as a result the resulting mesh has smooth edges and its shape is closer to the original geometry.

Important:

Only the nodes created in the adaption process (newly created nodes) are projected; the original nodes retain their positions.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Figure 25.5: Mesh Before Adaption

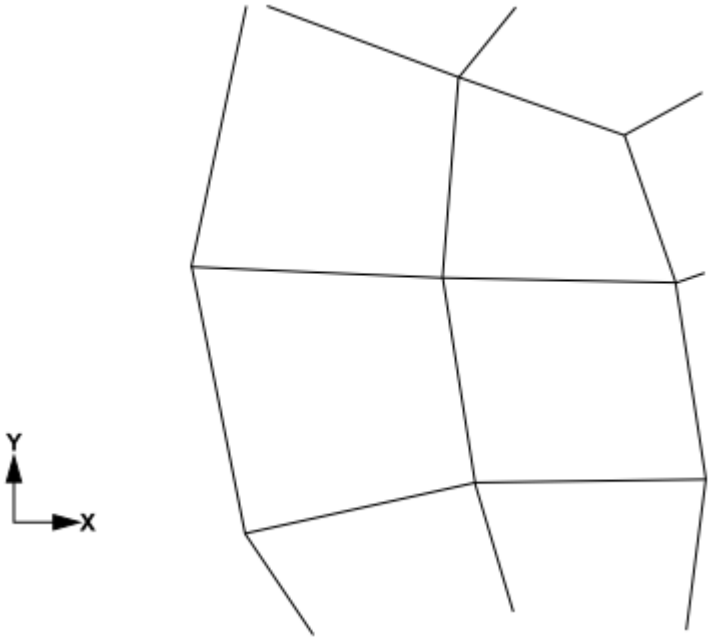
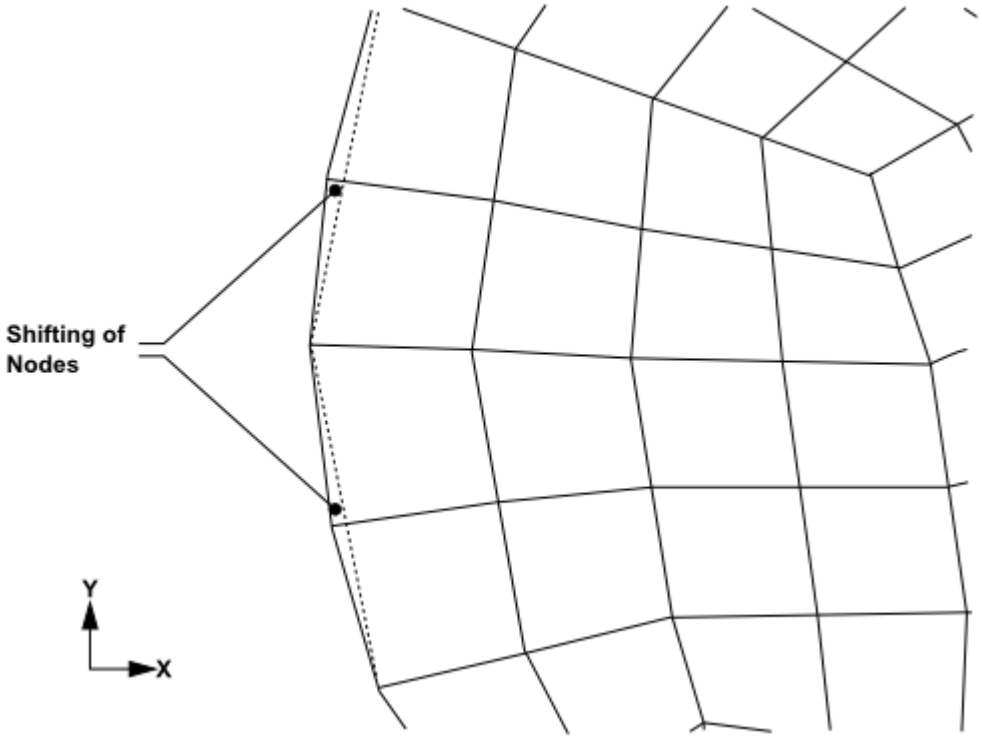


Figure 25.6: Projection of Nodes



When using the PUMA adaption method (which is the default for 3D), you can control the node projection / displacement using the following:

- auxiliary geometry definitions, which can be defined using a shape primitive, a user-defined function, or a surface mesh (as described in Managing Auxiliary Geometry Definitions in the *Fluent User's Guide*)

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

- an exponent that controls the degree to which the boundary displacement is also applied to the prismatic boundary layers adjacent to the surface (as described in Refining and Coarsening in the *Fluent User's Guide*)

For the hanging node adaption method, the following parameters control node projection and are specified in the Geometry Based Adaption Dialog Box.

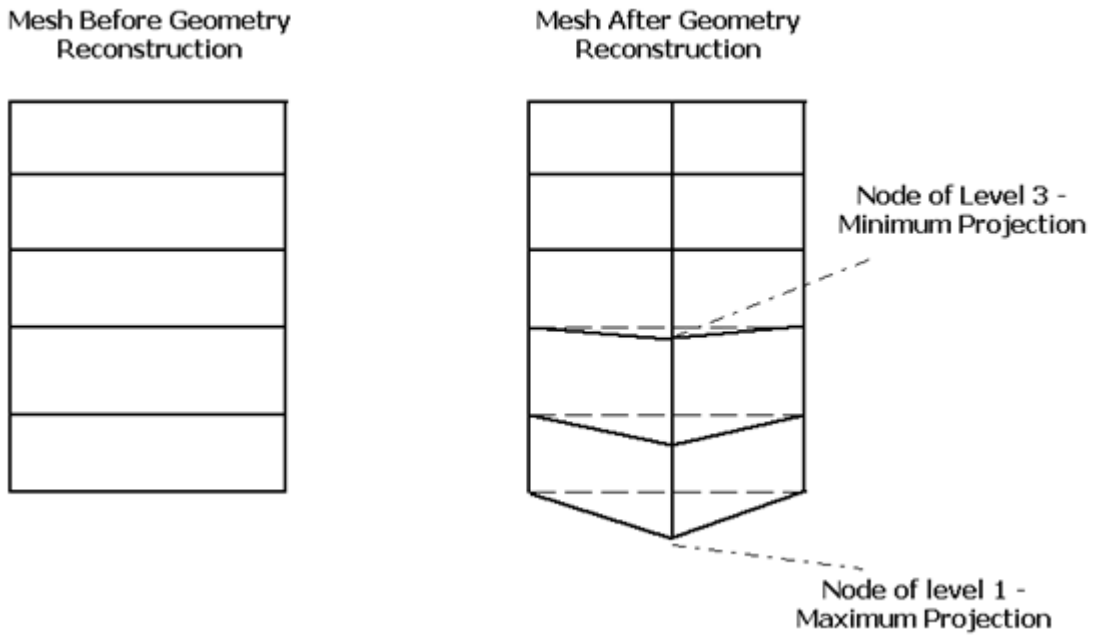
- **Levels of Projection Propagation**: This parameter allows you to specify the number of node layers across which node propagation should take place for geometry reconstruction. A value of 1 means only the nodes at the boundary will be projected, a value of 2 means the nodes at the boundary and the nodes in the next layer will be projected, and so on.

Note:

The nodes in the first level are projected by a maximum magnitude and the node in the last level are projected by a minimum magnitude. The magnitude of projection decreases gradually from the first level to the last level.

For example, a value of 3 for **Levels of Projection Propagation** means, the level 1 node is projected by maximum magnitude and level 3 node is projected by minimum magnitude. Figure 25.7: Levels Projection Propagation and Magnitude (p. 1027) illustrates the level of propagation and magnitude of projection of newly created nodes.

Figure 25.7: Levels Projection Propagation and Magnitude



- **Direction of Projection**: This parameter allows you to specify the direction, X, Y, or Z (for 3D), for node projection. If you do not specify any direction, the node projection takes place at the nearest point of the newly created node.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

- **Background Mesh**: This option allows you to use a fine surface mesh as a background mesh, which is then used to reconstruct the geometry. When you read the surface mesh, the node projection will take place based on the node positions of the background mesh.

  This option is useful when the mesh you want to adapt is very coarse and geometry is highly curved. In such cases, node projection, only by specifying the parameters may not result in a good quality mesh. However, you can also modify the propagation criteria by specifying the parameters.

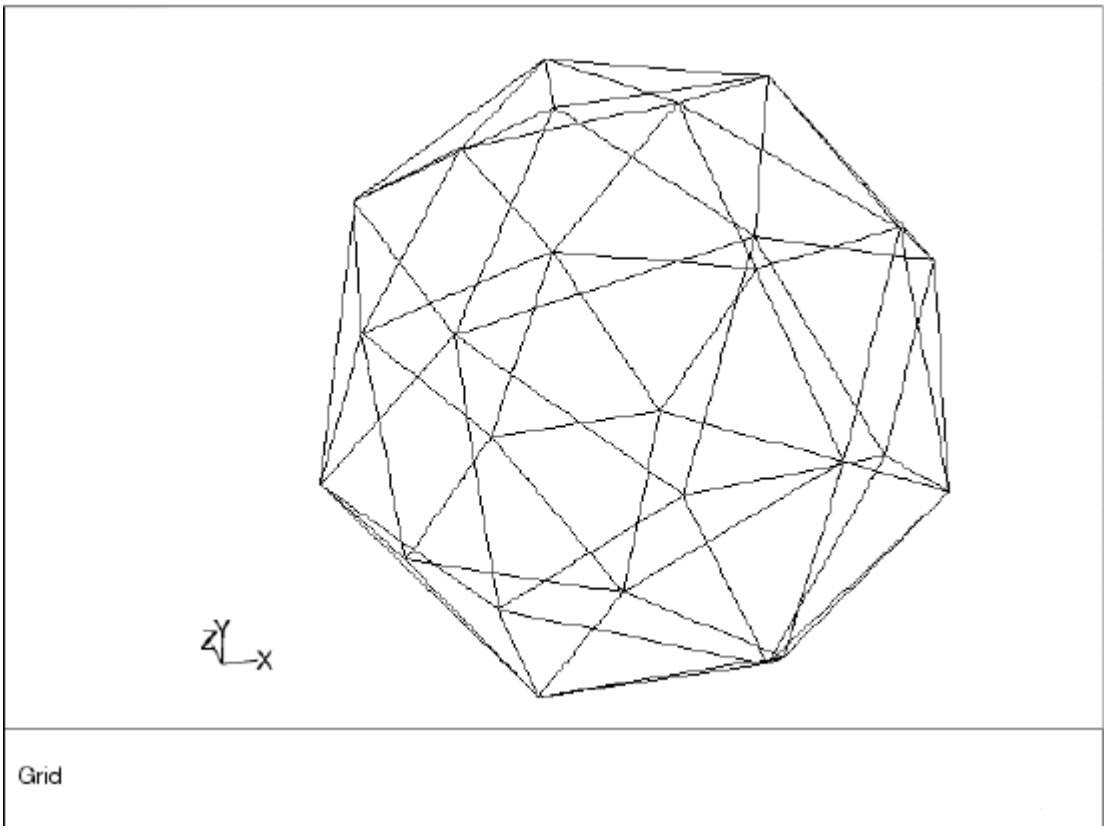
  Important:

  You can read only one surface mesh at a time. The various zones of the surface mesh will be listed in the **Background Mesh** drop-down list.

25.2.1.2. Example of Geometry-Based Adaption

Consider a mesh created for a spherical geometry. The initial mesh is very coarse, resulting in sharp corners (as in Figure 25.8: Coarse Mesh of a Sphere (p. 1028)). It does not represent the spherical geometry accurately. You can recover the original spherical geometry from this coarse mesh by using geometry-based adaption.

Figure 25.8: Coarse Mesh of a Sphere



If you adapt boundaries of the domain without enabling the **Reconstruct Geometry** option, the resulting mesh (see Figure 25.9: Adapted Mesh Without Geometry Reconstruction (p. 1029)) has a sufficient number of cells, but the boundary of the domain still contains sharp corners.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Boundary adaption only creates new nodes in between the existing nodes to increase the cell count of the mesh. Since it does not project the nodes, the shape of the mesh remains as it is.

If you adapt the boundary with the **Reconstruct Geometry** option enabled, the resulting mesh (Figure 25.10: Mesh after Geometry-Based Adaption (p. 1030)) has more cells and less sharp corners at the boundary. In addition, the newly created nodes are projected in a direction such that its shape is closer to the original geometry (that is, a sphere with smooth boundary).

Figure 25.9: Adapted Mesh Without Geometry Reconstruction

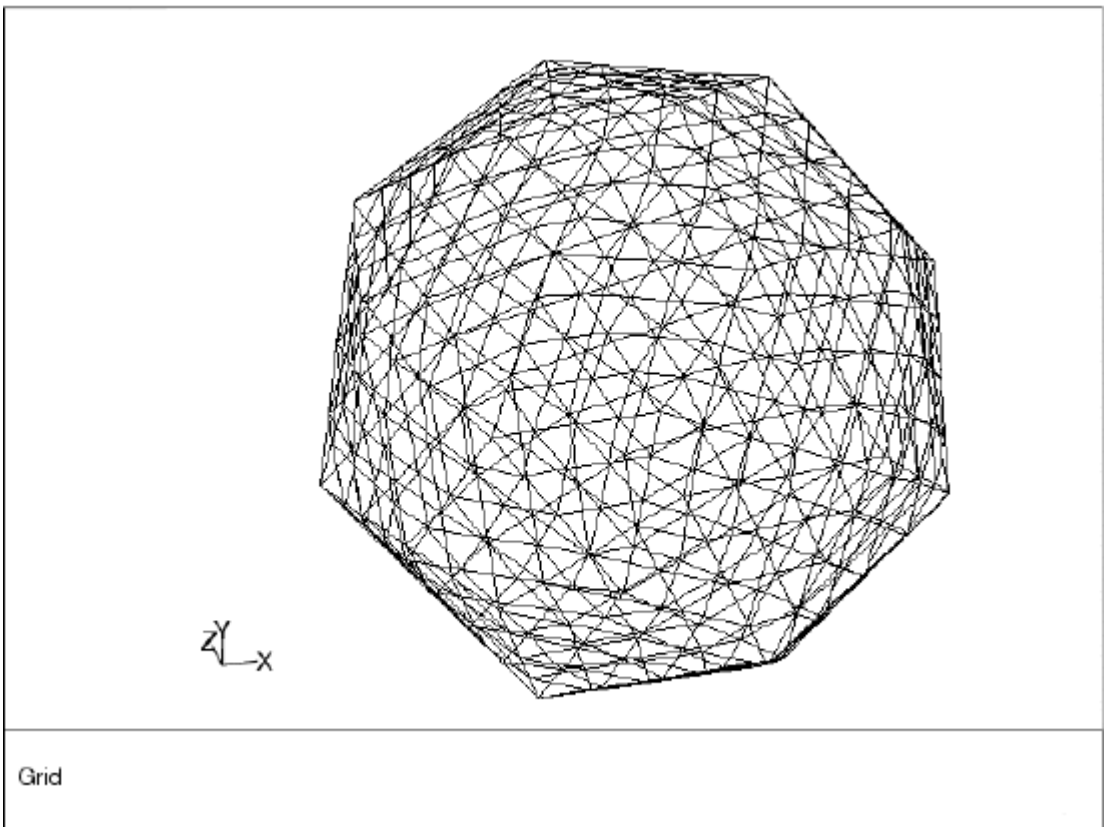
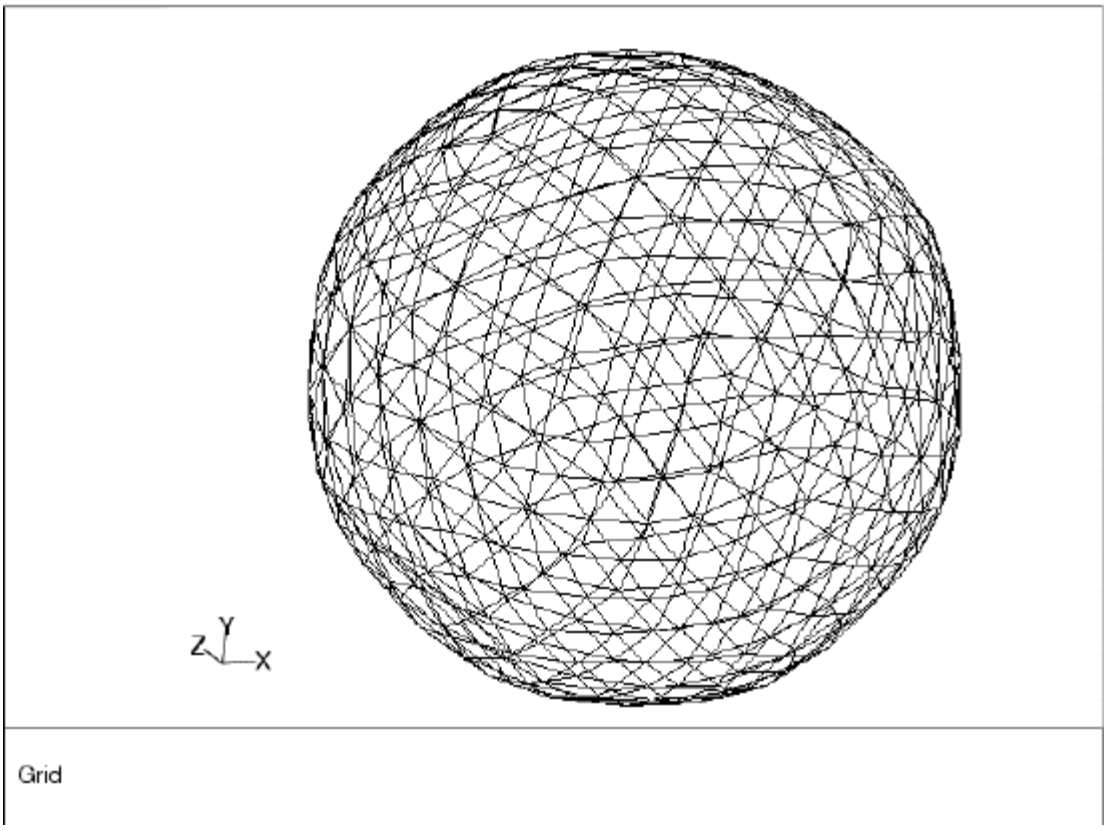


Figure 25.10: Mesh after Geometry-Based Adaption



Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information
of ANSYS, Inc. and its subsidiaries and affiliates.

Chapter 26: Reporting Alphanumeric Data

Ansys Fluent provides tools for computing and reporting integral quantities at surfaces and boundaries. These tools enable you to find the mass flow rate and heat transfer rate through boundaries, the forces and moments on boundaries, and the area, integral, flow rate, average, and mass average (among other quantities) on a surface or in a volume. In addition, you can print histograms of geometric and solution data, set reference values for the calculation of non-dimensional coefficients, and compute projected surface areas. You can also print or save a summary report of the models, boundary conditions, and solver settings in the current case.

This chapter describes some of the background behind Ansys Fluent's reporting features. These features are described in the following sections.

26.1. Fluxes Through Boundaries

26.2. Forces on Boundaries

26.3. Surface Integration

26.4. Volume Integration

26.5. Efficiency Calculation

Additional information about using these reporting tools can be found in Reporting Alphanumeric Data in the User's Guide. Reporting tools for the discrete phase are described in Postprocessing for the Discrete Phase in the User's Guide.

26.1. Fluxes Through Boundaries

This section describes the quantities that you can compute for boundaries. For more information about generating flux reports, see Generating a Flux Report in the User's Guide.

For selected boundary zones, you can compute the following quantities:

- The mass flow rate through a boundary is computed by summing the dot product of the density times the velocity vector and the area projections over the faces of the zone.
- The total heat transfer rate through a boundary is computed by summing the total heat transfer rate, $q=q_c+q_r$, over the faces, where q_c is the convective heat transfer rate and q_r is the radiation heat transfer rate. The computation of the heat transfer through the face depends on the specified boundary condition. For example, the conduction heat transfer on a constant-temperature wall face would be the product of the thermal conductivity with the dot product of the area projection and the temperature gradient. For flow boundaries, the total heat transfer rate is the flow rate of the conserved quantity. Depending on the models that are being used, the total heat transfer rate may

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

include the convective flow of sensible or total enthalpy, diffusive flux of energy, etc. Note that the reference temperature in all enthalpy calculations is always 298.15 K.

Important:

Note that in the case of moving walls with compressible flow, the pressure work is also added while reporting heat flux; this is also true with incompressible flow if the energy equation includes the pressure work (enabled through the `define/models/energy?` text command) or the viscous dissipation terms (enabled automatically for the density-based solver, or through the **Viscous Heating** option in the **Viscous Model** dialog box for the pressure-based solver).

- The radiation heat transfer rate through a boundary is computed by summing the radiation heat transfer rate q_r over the faces.

- The pressure work rate (I_{p}) is calculated for a fluid zone that undergoes normal zone motion relative to an adjacent solid zone, according to the following equation. This rate is included in the total heat transfer rate for boundary on the moving fluid zone (but not for the boundary of the adjacent solid zone).

$$I_{\mathrm{p}} = \int_A p\vec{u}_{\mathrm{m}} \cdot \vec{n}_{\mathrm{in}} dA \tag{26.1}$$

  where $\vec{u}_{\mathrm{m}}$ is the mesh velocity, dA is the surface element, $\vec{n}_{\mathrm{in}}$ is the normal vector of the surface element directed inside the fluid, p is the pressure, and A is the boundary surface area.

- The viscous work rate (I_{v}) through a boundary is computed by taking the integral of the product of viscous stress with velocity over the boundary with the boundary normal directed inside, according to the following equation. This rate is included in the total heat transfer rate with the pressure-based solver.

$$I_{\mathrm{v}} = \int_A \left(\vec{u} \cdot \overline{\overline{\tau}}\right) \cdot \vec{n}_{\mathrm{in}} dA \tag{26.2}$$

  where $\vec{u}$ is the fluid velocity, dA is the surface element, $\vec{n}_{\mathrm{in}}$ is the normal vector of the surface element directed inside the fluid, $\overline{\overline{\tau}}$ is the viscous stress tensor, and A is the boundary surface area.

For example, you might use flux reporting to compute the mass flow through a duct with pressure boundaries specified at the inlet and exit after running a simulation with these settings.

26.2. Forces on Boundaries

For selected wall zones, you can compute and report the forces along a specified vector, the moments about a specified center and along a specified axis, and the coordinates of the center of pressure. This feature is useful for reporting, for instance, aerodynamic quantities such as lift, drag, and moment coefficients, as well as the center of pressure for an airfoil.

For information about how Ansys Fluent computes forces, moments, and centers of pressure, see Computing Forces, Moments, and the Center of Pressure (p. 1033). Otherwise, for more information about

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

generating a report on forces, moments, or centers of pressure, see Generating a Force, Moment, or Center of Pressure Report in the User's Guide.

26.2.1. Computing Forces, Moments, and the Center of Pressure

The total force component along the specified force vector $\vec{a}$ on a wall zone is computed by summing the dot product of the pressure and viscous forces on each face with the specified force vector. The terms in this summation represent the pressure and viscous force components in the direction of the vector $\vec{a}$:

$$\underbrace{F_a}_{\substack{\text{total force}\\ \text{component}}} = \underbrace{\vec{a}\cdot\vec{F_p}}_{\substack{\text{pressure force}\\ \text{component}}} + \underbrace{\vec{a}\cdot\vec{F_v}}_{\substack{\text{viscous force}\\ \text{component}}} \tag{26.3}$$

where

$\vec{a}$ = specified force vector

$\vec{F_p}$ = pressure force vector

$\vec{F_v}$ = viscous force vector

In addition to the actual pressure, viscous, and total forces, the associated force coefficients are also computed for each of the selected wall zones, using the reference values (as described in Reference Values in the User's Guide). The force coefficient is defined as force divided by $\frac{1}{2}\rho v^2 A$, where ρ, v, and A are the density, velocity, and area. Finally, the net values of the pressure, viscous, and total forces and coefficients for all of the selected wall zones are also computed.

The total moment vector about a specified center A is computed by summing the cross products of the pressure and viscous force vectors for each face with the moment vector $\vec{r_{AB}}$, which is the vector from the specified moment center A to the force origin B (see Figure 26.1: Moment About a Specified Moment Center (p. 1034)). The terms in this summation represent the pressure and viscous moment vectors:

$$\underbrace{\vec{M_A}}_{\substack{\text{total}\\ \text{moment}}} = \underbrace{\vec{r_{AB}}\times\vec{F_P}}_{\substack{\text{pressure}\\ \text{moment}}} + \underbrace{\vec{r_{AB}}\times\vec{F_v}}_{\substack{\text{viscous}\\ \text{moment}}} \tag{26.4}$$

where

A = specified moment center

B = force origin

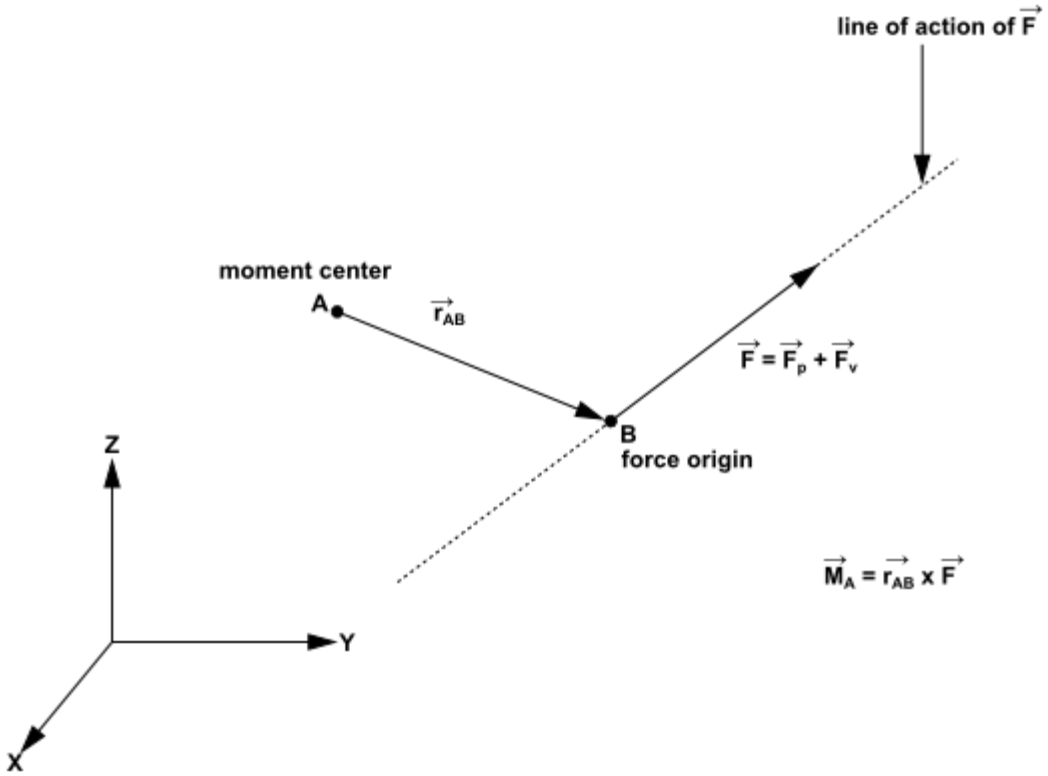
$\vec{r_{AB}}$ = moment vector

$\vec{F_P}$ = pressure force vector

$\vec{F_v}$ = viscous force vector

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Figure 26.1: Moment About a Specified Moment Center



Direction of the total moment vector follows the right hand rule for cross products.

Along with the actual components of the pressure, viscous, and total moments, the moment coefficients are computed for each of the selected wall zones, using the reference values (as described in Reference Values in the User's Guide). The moment coefficient is defined as the moment divided by $\frac{1}{2}\rho v^2 AL$, where ρ, v, A, and L are the density, velocity, area, and length. The coefficient values for the individual wall zones are also summed to yield the net values of the pressure, viscous, and total moments and coefficients for all of the selected wall zones.

Furthermore, the moments along a specified axis are computed. These moments, also known as torques, are defined as the dot product of a unit vector in the direction of the specified axis and the individual and net values of the pressure, viscous, and total moments and coefficients.

To reduce round-off error, a reference pressure is used to normalize the cell pressure for computation of the pressure force. For example, the net pressure force vector, acting on a wall zone, is computed as the vector sum of the individual force vectors for each cell face:

$$\vec{F_p} = \sum_{i=1}^{n}\left(p - p_{ref}\right)A\hat{n} \tag{26.5}$$

$$\vec{F_p} = \sum_{i=1}^{n} pA\hat{n} - p_{ref}\sum_{i=1}^{n} A\hat{n} \tag{26.6}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

where n is the number of faces, A is the area of the face, and $\hat{n}$ is the unit normal to the face.

Note:

For a multiphase flow, the phase-level forces, such as the pressure force $\vec{F_P}$ and viscous force $\vec{F_v}$, are multiplied by the volume fraction. The mixture-level forces and moments are the sums of the phase-level forces and moments, respectively. For example, the pressure force for a given phase is calculated as:

$$\vec{F}_{p,phase} = \sum_{i=1}^{n} \alpha \left(p - p_{ref} \right) A\hat{n} \tag{26.7}$$

and the pressure force at the mixture level is calculated as:

$$\vec{F}_{p,mixture} = \sum_{j=1}^{N} \sum_{i=1}^{n} \alpha_j \left(p - p_{ref} \right) A\hat{n} = \sum_{i=1}^{n} \left(p - p_{ref} \right) A\hat{n} \tag{26.8}$$

where N is the number of phases, and α is the volume fraction.

For a 2D shape, the resulting force due to pressure and viscous stresses is applied along a line (parallel to the resulting force). The center of pressure is the intersecting point of this line with a user-specified reference line (for example, the chord line is generally selected for a 2D airfoil). The resultant moment about this point is then zero.

For a general body in 3D, the pressure and viscous wall stress distribution can be represented by a force (and its application axis) and a moment about a moment center. In general, the problem of finding the translation of the application axis that zeroes the moment does not have a solution. For certain symmetric geometries (for example a non spinning missile), however, this solution exists. In such cases, the center of pressure is usually defined as the intersection of the application axis of the resulting force with a user-specified reference plane.

Ansys Fluent calculates the center of pressure as follows. For a generic moment center and axes, the resultant moment is expressed as

$$\begin{aligned} M_x &= F_z Y - F_y Z \\ M_y &= F_x Z - F_z X \\ M_z &= F_y X - F_x Y \end{aligned} \tag{26.9}$$

Important:

The forces and moments referenced in Equation 26.9 (p. 1035) for computing the moment center and axis are only pressure forces and moments and they do not include viscous forces.

In 3D, zeroing two of these equations and using the equation of the user-specified (constraining) reference plane, the intersection point between the application axis and the specified reference plane can be obtained. In 2D, only the last equation in Equation 26.9 (p. 1035) is used in combination with the user-specified reference line to calculate the center of pressure.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

26.3. Surface Integration

You can compute the area or mass flow rate, or the integral, standard deviation, flow rate, volume flow rate, area-weighted average, mass-weighted average, custom vector-weighted average, custom vector-based flux, custom vector flux, sum, facet average, facet maximum, facet minimum, uniformity index (weighted by area or mass), vertex average, vertex minimum, and vertex maximum for a selected field variable on selected surfaces in the domain. These surfaces are sets of data points created by Ansys Fluent for each of the zones in your model, or defined by you using the methods described in Creating Surfaces and Cell Registers for Displaying and Reporting Data in the User's Guide, or by using the Text User Interface, described in Text User Interface (TUI) in the User's Guide.

The following list provides information on the data reported for the various surface integrals:

- For the vertex average, vertex maximum, and the vertex minimum, Ansys Fluent reports the node values of the selected variable on the selected surface.
- For mass flow rate, volume flow rate, and flow rate, Ansys Fluent reports the rate. Of these, flow rate is the only one associated with a selected variable. The values used in the computation depend on the kind of surface selected:
  - The face flux values are used for face zone surfaces.
  - The cell values are used for postprocessing surfaces. See Cell Values in the User's Guide for further details.
- For all other surface integrals, Ansys Fluent reports the integral, using values that are appropriate for the particular surface:
  - For face zone surfaces, the face values are used when they are available, that is, when they are calculated by the solver or specified as a boundary condition. Otherwise, the cell values are used. The cell value for non-internal faces (that is, faces that only have `c0` and no `c1`) is the `c0` value. The cell value for internal faces (that is, faces that have `c0` and `c1`) is the average of the `c0` and `c1` values.
  - For postprocessing surfaces, the cell values are used. See Cell Values in the User's Guide for further details.

Example uses of several types of surface integral reports are given below:

- Area: You can compute the area of a velocity inlet zone, and then estimate the velocity from the mass flow rate:

$$v=\frac{\dot{m}}{\rho A} \tag{26.10}$$

- Area-weighted average: You can find the average value on a solid surface, such as the average heat flux on a heated wall with a specified temperature.
- Mass average: You can find the average value on a surface in the flow, such as average enthalpy at a velocity inlet.
- Mass flow rate: You can compute the mass flow rate through a velocity inlet zone, and then estimate the velocity from the area, as described above.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

- Flow rate: To calculate the heat transfer rate through a surface, you can calculate the flow rate of enthalpy.
- Integral: You can use integrals for more complex calculations, which may involve the use of the Custom Field Function Calculator Dialog Box, described in Custom Field Functions in the User's Guide, to calculate a function that requires integral computations (for example, swirl number).
- Standard deviation: You can find the standard deviation of a specified field variable on a surface, such as enthalpy, viscosity, and velocity.
- Volume flow rate: This will report the volume flow rate through the specified surface.

For information about how Ansys Fluent computes surface integrals, see Computing Surface Integrals (p. 1037). Otherwise, for more information about generating a surface integral report, see Generating a Surface Integral Report in the User's Guide.

26.3.1. Computing Surface Integrals

26.3.1.1. Area

The area of a surface is computed by summing the areas of the facets that define the surface. Facets on a surface are either triangular or quadrilateral in shape.

$$\int dA = \sum_{i=1}^{n} |A_i| \tag{26.11}$$

26.3.1.2. Integral

An integral on a surface is computed by summing the product of the facet area and the selected field variable facet value, such as density or pressure. For details on the computation of the facet values, see Surface Integration (p. 1036).

26.3.1.3. Area-Weighted Average

The area-weighted average of a quantity is computed by dividing the summation of the product of the selected field variable and facet area by the total area of the surface:

$$\frac{1}{A}\int \phi dA = \frac{1}{A}\sum_{i=1}^{n} \phi_i |A_i| \tag{26.12}$$

26.3.1.4. Custom Vector Based Flux

The custom vector-based flux of a quantity is computed by summing the value of the selected field variable multiplied by the dot product of the custom velocity vector and the area vector.

$$\int \Phi \vec{v}_c \cdot d\vec{A} = \sum_{i=1}^{n} \Phi_i \vec{v}_{c_i} \cdot \vec{A}_i \tag{26.13}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

26.3.1.5. Custom Vector Flux

The custom vector flux is computed by summing the dot product of the custom velocity vector and the area vector.

$$\int \vec{v}_c \cdot d\vec{A} = \sum_{i=1}^{n} \vec{v}_{c_i} \cdot \vec{A}_i \tag{26.14}$$

26.3.1.6. Custom Vector Weighted Average

The custom vector-weighted average of a quantity is computed by dividing the summation of the value of the selected field variable multiplied by the absolute value of the dot product of the custom velocity vectors and the area vectors by the summation of the absolute value of the dot product of the custom velocity vectors and the area vectors.

$$\frac{\int \Phi \left| \vec{v}_c \cdot d\vec{A} \right|}{\int \left| \vec{v}_c \cdot d\vec{A} \right|} = \frac{\sum_{i=1}^{n} \Phi_i \left| \vec{v}_{c_i} \cdot \vec{A}_i \right|}{\sum_{i=1}^{n} \left| \vec{v}_{c_i} \cdot \vec{A}_i \right|} \tag{26.15}$$

26.3.1.7. Flow Rate

The flow rate of a quantity through a surface is computed by summing the value of the selected field variable multiplied by the density and the dot product of the facet area vector and the facet velocity vector.

$$\int \phi \rho \vec{v} \cdot d\vec{A} = \sum_{i=1}^{n} \phi_i \rho_i \vec{v}_i \cdot \vec{A}_i \tag{26.16}$$

Note:

The density, ρ_i, used in Equation 26.16 (p. 1038) is the mixture density.

26.3.1.8. Mass Flow Rate

The mass flow rate through a surface is computed by summing the value of the selected field variable multiplied by the density and the dot product of the facet area vector and the facet velocity vector.

$$\int \rho \vec{v} \cdot d\vec{A} = \sum_{i=1}^{n} \rho_i \vec{v}_i \cdot \vec{A}_i \tag{26.17}$$

Note:

For multiphase cases, the density, ρ_i, used in Equation 26.17 (p. 1038) is the phase density.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

26.3.1.9. Mass-Weighted Average

The mass-weighted average of a quantity is computed by dividing the summation of the value of the selected field variable multiplied by the absolute value of the dot product of the facet area and momentum vectors by the summation of the absolute value of the dot product of the facet area and momentum vectors (surface mass flux):

$$\frac{\int \phi\rho\left|\vec{v}\cdot d\vec{A}\right|}{\int \rho\left|\vec{v}\cdot d\vec{A}\right|} = \frac{\sum_{i=1}^{n}\phi_i\rho_i\left|\vec{v}_i\cdot\vec{A}_i\right|}{\sum_{i=1}^{n}\rho_i\left|\vec{v}_i\cdot\vec{A}_i\right|} \tag{26.18}$$

Note:

Ansys Fluent uses bulk-based fluxes for computing mass-weighted averages. With bulk-density represented as $\hat{\rho}$ and bulk-velocity represented as $\hat{v}$, the discrete mass-weighted average of a quantity ϕ is given by:

$$\frac{\sum_{i=1}^{n}\phi_i\hat{\rho}_i\left|\hat{v}_i\cdot A_i\right|}{\sum_{i=1}^{n}\hat{\rho}_i\left|\hat{v}_i\cdot A_i\right|} \tag{26.19}$$

where the index, i, goes over the face points. Since $\hat{\rho}_i$ is always positive, it can be re-written as:

$$\frac{\sum_{i=1}^{n}\phi_i\left|\hat{\rho}_i\hat{v}_i\cdot A_i\right|}{\sum_{i=1}^{n}\left|\hat{\rho}_i\hat{v}_i\cdot A_i\right|} \tag{26.20}$$

The computation of $\hat{\rho}_i\hat{v}_i$ depends on the model. For single-phase cases, it is:

$$\hat{\rho}_i\hat{v}_i = \psi_i\rho_i v_i \tag{26.21}$$

where ψ is the porosity of the medium. For multiphase cases, it is:

$$\hat{\rho}_i\hat{v}_i = \psi_i\sum_j \alpha_{ji}\rho_{ji}v_{ji} \tag{26.22}$$

where the summation of j goes over the phases, and α_j is the j^{th} phase-fraction computed from the appropriate model.

Important:

For multiphase flows, the mass-weighted average computation is only recommended for mixture-level variables, not phase-level ones.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

26.3.1.10. Sum of Field Variable

The sum of a specified field variable on a surface is computed by summing the value of the selected variable at each facet:

$$\sum_{i=1}^{n} \phi_i \tag{26.23}$$

Note:

For multiphase cases, the density, ρ_i, used in Equation 26.23 (p. 1040) is the mixture density.

26.3.1.11. Facet Average

The facet average of a specified field variable on a surface is computed by dividing the summation of the facet values of the selected variable by the total number of facets. See Node, Cell, and Facet Values in the User's Guide for definitions of facet values.

$$\frac{\sum_{i=1}^{n} \phi_i}{n} \tag{26.24}$$

26.3.1.12. Facet Minimum

The facet minimum of a specified field variable on a surface is the minimum facet value of the selected variable on the surface. See Node, Cell, and Facet Values in the User's Guide for definitions of facet values.

26.3.1.13. Facet Maximum

The facet maximum of a specified field variable on a surface is the maximum facet value of the selected variable on the surface. See Node, Cell, and Facet Values in the User's Guide for definitions of facet values.

26.3.1.14. Vertex Average

The vertex average of a specified field variable on a surface is computed by dividing the summation of the vertex values of the selected variable by the total number of vertices. See Node, Cell, and Facet Values in the User's Guide for definitions of vertex values.

$$\frac{\sum_{i=1}^{n} \phi_i}{n} \tag{26.25}$$

26.3.1.15. Vertex Minimum

The vertex minimum of a specified field variable on a surface is the minimum vertex value of the selected variable on the surface. See Node, Cell, and Facet Values in the User's Guide for definitions of vertex values.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

26.3.1.16. Vertex Maximum

The vertex maximum of a specified field variable on a surface is the maximum vertex value of the selected variable on the surface. See Node, Cell, and Facet Values in the User's Guide for definitions of vertex values.

26.3.1.17. Standard-Deviation

The standard deviation of a specified field variable on a surface is computed using the mathematical expression below:

$$\sqrt{\frac{\sum_{i=1}^{n}(\phi-\phi_0)^2}{n}} \tag{26.26}$$

where ϕ is the cell value of the selected variables at each facet, ϕ_0 is the mean of ϕ

$$\phi_0=\frac{\sum_{i=1}^{n}\phi}{n}$$

and n is the total number of facets. See Node, Cell, and Facet Values in the User's Guide for definitions of facet values.

26.3.1.18. Uniformity Index

The uniformity index represents how a specified field variable varies over a surface, where a value of 1 indicates the highest uniformity. The uniformity index can be weighted by area or mass: the area-weighted uniformity index captures the variation of the quantity (for example, the species concentration), whereas the mass-weighted uniformity index captures the variation of the flux (for example, the species flux).

The area-weighted uniformity index (γ_a) of a specified field variable ϕ is calculated using the following equation:

$$\gamma_a=1-\frac{\sum_{i=1}^{n}\left[\left(\left|\phi_i-\overline{\phi}_a\right|\right)A_i\right]}{2\left|\overline{\phi}_a\right|\sum_{i=1}^{n}A_i} \tag{26.27}$$

where i is the facet index of a surface with n facets, and $\overline{\phi}_a$ is the average value of the field variable over the surface:

$$\overline{\phi}_a=\frac{\sum_{i=1}^{n}\phi_i A_i}{\sum_{i=1}^{n}A_i} \tag{26.28}$$

The equation for the mass-weighted uniformity index (γ_m) is different than Equation 26.27 (p. 1041) in that it incorporates flux terms:

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$$\gamma_m = 1 - \frac{\sum_{i=1}^{n} \left[\left(\left| \phi_i - \bar{\phi}_m \right| \right) \left(\left| \rho_i \vec{v}_i \cdot \vec{A}_i \right| \right) \right]}{2 \left| \bar{\phi}_m \right| \sum_{i=1}^{n} \left[\left| \rho_i \vec{v}_i \cdot \vec{A}_i \right| \right]} \tag{26.29}$$

where $\bar{\phi}_m$ is the average flux of the field variable through the surface:

$$\bar{\phi}_m = \frac{\sum_{i=1}^{n} \left[\phi_i \left(\left| \rho_i \vec{v}_i \cdot \vec{A}_i \right| \right) \right]}{\sum_{i=1}^{n} \left[\left| \rho_i \vec{v}_i \cdot \vec{A}_i \right| \right]} \tag{26.30}$$

26.3.1.19. Volume Flow Rate

The volume flow rate through a surface is computed by summing the value of the facet area vector multiplied by the facet velocity vector:

$$\int \vec{v} \cdot d\vec{A} = \sum_{i=1}^{n} \vec{v}_i \cdot \vec{A}_i \tag{26.31}$$

26.4. Volume Integration

The volume, sum, maximum, minimum, volume integral, volume-weighted average, mass integral, and mass-weighted average can be obtained for a selected field variable in selected cell zones in the domain.

Example uses of the different types of volume integral reports are given below:

- **Volume**: You can compute the total volume of a fluid region, or the volume of a phase within the fluid region.
- **Sum**: You can add up the discrete-phase mass or energy sources to determine the net transfer from the discrete phase. You can also sum user-defined sources of mass or energy.
- **Sum*2Pi**: In a 2D axisymmetric model, you can add up the discrete-phase mass or energy sources to determine the net transfer from the discrete phase in the entire revolved domain. You can also sum user-defined sources of mass or energy. This report type is only available in 2D axisymmetric cases.
- **Maximum**: The maximum value of the selected variable among all cells in the selected zone.
- **Minimum**: The minimum value of the selected variable among all cells in the selected zone.
- **Volume Integral**: For quantities that are stored per unit volume, you can use volume integrals to determine the net value (for example, integrate density to determine mass).
- **Volume-Average**: You can obtain volume averages of mass sources, energy sources, or discrete-phase exchange quantities.
- **Mass Integral**: You can determine the total mass of a particular species by integrating its mass fraction.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

- **Mass**: You can determine the total mass of the mixture phase or an individual phase in the selected fluid zone.
- **Mass-Average**: You can find the average value (such as average temperature) in a fluid zone.

For information about how Ansys Fluent computes volume integrals, see Computing Volume Integrals (p. 1043). Otherwise, for more information about generating a volume integral report, see Generating a Volume Integral Report in the User's Guide.

26.4.1. Computing Volume Integrals

26.4.1.1. Volume

The volume of a fluid phase in a cell zone is computed by summing the product of the phase volume fraction and cell volume for the cells that make up the zone:

$$\int \alpha_p dV = \sum_{i=1}^{n} \alpha_{p_i} |V_i| \tag{26.32}$$

Note:

For single-phase flow, or for the mixture phase in a multiphase flow, $\alpha_{p_i}=1$ and the volume of the cell zone is recovered.

26.4.1.2. Sum

The sum of a specified field variable in a cell zone is computed by summing the value of the selected variable at each cell in the selected zone:

$$\sum_{i=1}^{n} \phi_i \tag{26.33}$$

26.4.1.3. Sum*2Pi

The sum*2pi of a specified field variable in a cell zone of a 2D axisymmetric model is computed by summing the value of the selected variable at each cell in the selected zone and multiplying it by 2π:

$$2\pi \sum_{i=1}^{n} \phi_i \tag{26.34}$$

26.4.1.4. Volume Integral

A volume integral is computed by summing the product of the cell volume and the selected field variable:

$$\int \phi dV = \sum_{i=1}^{n} \phi_i |V_i| \tag{26.35}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

26.4.1.5. Volume-Weighted Average

The volume-weighted average of a quantity is computed by dividing the summation of the product of the selected field variable and cell volume by the total volume of the cell zone:

$$\frac{1}{V}\int \phi dV = \frac{1}{V}\sum_{i=1}^{n} \phi_i |V_i| \tag{26.36}$$

26.4.1.6. Mass-Weighted Integral

The mass-weighted integral is computed by summing the product of density, cell volume, and the selected field variable:

$$\int \phi \rho dV = \sum_{i=1}^{n} \phi_i \rho_i |V_i| \tag{26.37}$$

Note:

For multiphase cases, the density, ρ_i, used in Equation 26.37 (p. 1044) is the mixture density.

26.4.1.7. Mass

The mass of a particular phase is computed by summing the product of phase density, cell volume, and phase volume fraction:

$$\int \alpha_p \rho_p dV = \sum_{i=1}^{n} \alpha_{p_i} \rho_{p_i} |V_i| \tag{26.38}$$

26.4.1.8. Mass-Weighted Average

The mass-weighted average of a quantity is computed by dividing the summation of the product of density, cell volume, and the selected field variable by the summation of the product of density and cell volume:

$$\frac{\int \phi \rho dV}{\int \rho dV} = \frac{\sum_{i=1}^{n} \phi_i \rho_i |V_i|}{\sum_{i=1}^{n} \rho_i |V_i|} \tag{26.39}$$

Note:

For multiphase cases, the density, ρ_i, used in Equation 26.39 (p. 1044) is the mixture density.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

26.5. Efficiency Calculation

For selected surfaces, you can compute and report the values of isentropic and polytropic efficiencies. For information about supported surfaces and available tools for efficiency calculation, see Efficiency Calculations in the *Fluent User's Guide*. For information about how Ansys Fluent computes efficiency values, see the following sections:

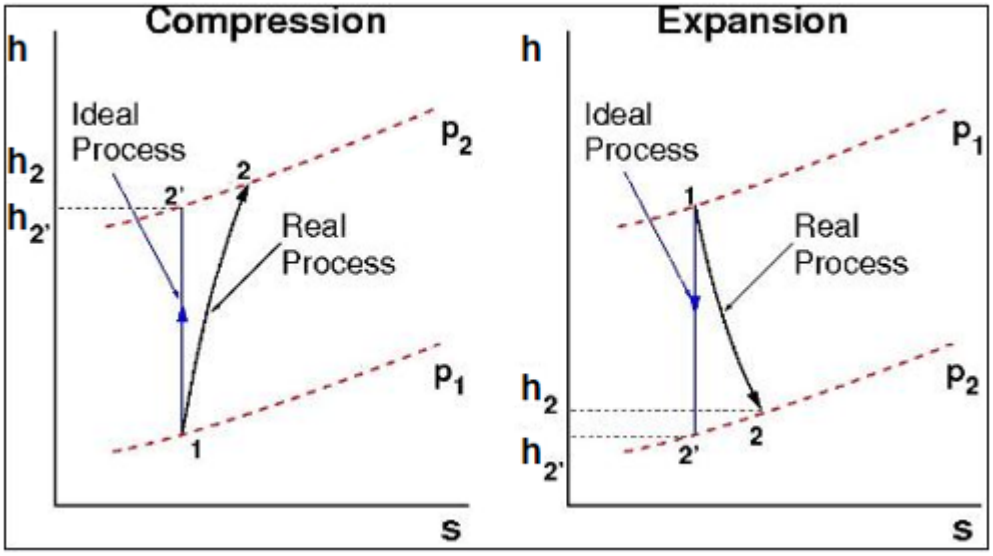
26.5.1. Isentropic Efficiency

26.5.2. Polytropic Efficiency

26.5.1. Isentropic Efficiency

The efficiency of any device is never perfect due to both fluid dynamic and thermodynamic losses within the entire system. The losses appear as a difference between actual values from those that would be expected if the process was perfectly isentropic. Consider the processes drawn on the T-s diagrams shown in Figure 26.2: T-s Diagrams for Compression and Expansion Processes (p. 1045).

Figure 26.2: T-s Diagrams for Compression and Expansion Processes



Three efficiency calculation options are available for either process: **Total to Total**, **Total to Static**, and **Static to Static**.

For an expansion process (for example, in a steam turbine), these options are defined by the following ratios:

$$\eta_{E,TT}=\frac{h_{01}-h_{02}}{h_{01}-h_{02'}} \tag{26.40}$$

$$\eta_{E,TS}=\frac{h_{01}-h_{02}}{h_{01}-h_{2'}} \tag{26.41}$$

$$\eta_{E,SS}=\frac{h_{01}-h_{2}}{h_{01}-h_{2'}} \tag{26.42}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

where h_{01} and h_{02} are the mass-averaged values of the total enthalpy at points 1 and 2, respectively, and h_2 is the mass-averaged value of the static enthalpy at point 2. In addition, the isentropic total enthalpy $h_{02'}$ and isentropic static enthalpy $h_{2'}$ are required. They are evaluated from the state equation:

$$h_{2'}=h\left(s_1,p_2\right) \tag{26.43}$$

$$h_{02'}=h\left(s_1,p_{02}\right) \tag{26.44}$$

where the entropy s_1 is taken at point 1, by default, as the mass-averaged value, p_2 is the area-averaged value of the static pressure at point 2, and p_{02} is the mass-flow averaged value of the total pressure at point 2.

For a compression process (for example, in a gas compressor), the ratios Equation 26.40 (p. 1045), Equation 26.41 (p. 1045), and Equation 26.42 (p. 1045) are inversed and read as:

$$\eta_{C,TT}=\frac{h_{01}-h_{02'}}{h_{01}-h_{02}} \tag{26.45}$$

$$\eta_{C,TS}=\frac{h_{01}-h_{2'}}{h_{01}-h_{02}} \tag{26.46}$$

$$\eta_{E,SS}=\frac{h_{01}-h_{2'}}{h_{01}-h_2} \tag{26.47}$$

The **Total to Total** and **Total to Static** option are most useful for devices where work is either performed on the fluid (for example, a pump or compressor) or extracted from the fluid (for example, a turbine). The **Static to Static** option is useful for evaluating losses in devices that perform no work on the fluid (for example, diffusers and nozzles).

26.5.2. Polytropic Efficiency

The isentropic efficiency has one significant drawback in that there is no way to separate the fluid dynamic losses from the total (fluid dynamic + thermodynamic) losses. This means that devices having different pressure ratios will have different isentropic efficiencies even though they may both be of similar fluid dynamic quality. An example would be two compressors of different pressure ratios. The higher-pressure ratio compressor will have a lower isentropic efficiency because of thermodynamic losses. This property can make it difficult to comparatively evaluate different compressor designs. A similar argument applies to turbine design.

To work around this drawback, the assumed "ideal" path followed by the process does not have to be isentropic. Instead, one can evaluate polytropic efficiency by following a path along a line of constant efficiency. Aungier [31] (p. 1054) discusses how a constant efficiency path on a T-s diagram can be approximated by the following equation:

$$T\frac{ds}{dT}=A \tag{26.48}$$

This equation can be rearranged to give the following two relationships:

$$Tds=AdT,\ ds=\frac{AdT}{T} \tag{26.49}$$

The entropy change along this path is evaluated by integrating the last expression:

$$s_2-s_1=A\ln\left(T_{02}/T_{01}\right) \tag{26.50}$$

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Rearranging this expression gives the value of the constant A along the given path:

$$A=\frac{s_2-s_1}{\ln\left(T_{02}/T_{01}\right)} \tag{26.51}$$

To evaluate the polytropic efficiency, evaluate the polytropic enthalpy change along the alternative path defined by the path equation. Starting with the second law:

$$Tds=dh-vdp \tag{26.52}$$

The vdp term is the polytropic work, or enthalpy change, and is what must be solved for. First, the Tds term is substituted using Equation 26.49 (p. 1046) to give:

$$AdT=Tds=dh-vdp \tag{26.53}$$

Integrating this equation along the polytropic path gives:

$$vdp=\left(h_{02}-h_{01}\right)-A\left(T_{02}-T_{01}\right) \tag{26.54}$$

The polytropic enthalpy can be obtained from the polytropic enthalpy change vdp simply by omitting the subtraction of h_{01}. Also, substituting for A gives the final form of the polytropic total enthalpy $h_{02'}^{poly}$:

$$h_{02'}^{poly}=h_{02}-\frac{\left(s_2-s_1\right)\left(T_{02}-T_{01}\right)}{\ln\left(T_{02}/T_{01}\right)} \tag{26.55}$$

Polytropic static enthalpy $h_{2'}^{poly}$ is evaluated by replacing total quantities at point 2 with static quantities:

$$h_{2'}^{poly}=h_2-\frac{\left(s_2-s_1\right)\left(T_2-T_{01}\right)}{\ln\left(T_2/T_{01}\right)} \tag{26.56}$$

Using $h_{2'}^{poly}$ and $h_{02'}^{poly}$ instead of $h_{2'}$ and $h_{02'}$ in formulas Equation 26.40 (p. 1045)-Equation 26.42 (p. 1045) and Equation 26.45 (p. 1046)-Equation 26.47 (p. 1046), the corresponding value of the polytropic efficiency can be evaluated.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information
of ANSYS, Inc. and its subsidiaries and affiliates.

Appendix A. Nomenclature

A	Area (m^2, ft^2)
$\vec{a}$	Acceleration (m/s^2, ft/s^2)
a	Local speed of sound (m/s, ft/s)
c	Concentration (mass/volume, moles/volume)
C_D	Drag coefficient, defined different ways (dimensionless)
c_p, c_v	Heat capacity at constant pressure, volume (J/kg-K, Btu/lb_m- °F)
d	Diameter; d_p,D_p, particle diameter (m, ft)
D_H	Hydraulic diameter (m, ft)
$_{ij}$,D	Mass diffusion coefficient (m^2/s, ft^2/s)
E	Total energy, activation energy (J, kJ, cal, Btu)
f	Mixture fraction (dimensionless)
$\vec{F}$	Force vector (N, lb_f)
F_D	Drag force (N, lb_f)
$\vec{g}$	Gravitational acceleration (m/s^2, ft/s^2); standard values = 9.80665 m/s^2, 32.1740 ft/s^2
Gr	Grashof number ≡ ratio of buoyancy forces to viscous forces (dimensionless)
H	Total enthalpy (energy/mass, energy/mole)
h	Heat transfer coefficient (W/m^2-K, Btu/ft^2-h- °F)
h	Species enthalpy; h^0, standard state enthalpy of formation (energy/mass, energy/mole)
I	Radiation intensity (energy per area of emitting surface per unit solid angle)
J	Mass flux; diffusion flux (kg/m^2-s, lb_m/ft^2-s)
K	Equilibrium constant = forward rate constant/backward rate constant (units vary)
k	Kinetic energy per unit mass (J/kg, Btu/lb_m)
k	Reaction rate constant, for example, k_1, k_{-1}, $k_{f,r}$, $k_{b,r}$ (units vary)
k	Thermal conductivity (W/m-K, Btu/ft-h- °F)
k_B	Boltzmann constant (1.38×10^{-23} J/molecule-K)
k, k_c	Mass transfer coefficient (units vary); also K, K_c
ℓ,l,L	Length scale (m, cm, ft, in)
Le	Lewis number ≡ ratio of thermal diffusivity to mass diffusivity (dimensionless)
m	Mass (g, kg, lb_m)

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

$\dot{m}$	Mass flow rate (kg/s, lb_m/s)
M_w	Molecular weight (kg/kmol, $\mathrm{lb}_m/\mathrm{lb}_m$mol)
M	Mach number ≡ ratio of fluid velocity magnitude to local speed of sound (dimensionless)
Nu	Nusselt number ≡ dimensionless heat transfer or mass transfer coefficient (dimensionless); usually a function of other dimensionless groups
p	Pressure (Pa, atm, mm Hg, $\mathrm{lb}_f/\mathrm{ft}^2$)
Pe	Peclet number $\equiv Re \times Pr$ for heat transfer, and $\equiv Re \times Sc$ for mass transfer (dimensionless)
Pr	Prandtl number ≡ ratio of momentum diffusivity to thermal diffusivity (dimensionless)
Q	Flow rate of enthalpy (W, Btu/h)
q	Heat flux (W/m^2, Btu/ft^2-h)
R	Gas-law constant (8.31447 $\times 10^3$ J/kmol-K, 1.98588 Btu/lb_mmol- °F)
r	Radius (m, ft)
$\mathcal{R}$	Reaction rate (units vary)
Ra	Rayleigh number $\equiv Gr \times Pr$; measure of the strength of buoyancy-induced flow in natural (free) convection (dimensionless)
Re	Reynolds number ≡ ratio of inertial forces to viscous forces (dimensionless)
S	Total entropy (J/K, J/kmol-K, Btu/lb_mmol- °F)
s	Species entropy; s^0, standard state entropy (J/kmol-K, Btu/lb_mmol- °F)
Sc	Schmidt number ≡ ratio of momentum diffusivity to mass diffusivity (dimensionless)
S_{ij}	Mean rate-of-strain tensor (s^{-1})
T	Temperature (K, °C, °R, °F)
t	Time (s)
U	Free-stream velocity (m/s, ft/s)
u, v, w	Velocity magnitude (m/s, ft/s); also written with directional subscripts (for example, v_x, v_y, v_z, v_r)
V	Volume (m^3, ft^3)
$\vec{v}$	Overall velocity vector (m/s, ft/s)
We	Weber number ≡ ratio of aerodynamic forces to surface tension forces (dimensionless)
X	Mole fraction (dimensionless)
Y	Mass fraction (dimensionless)
α	Permeability, or flux per unit pressure difference (L/m^2-h-atm, ft^3/ft^2-h-(lb_f/ft^2))
α	Thermal diffusivity (m^2/s, ft^2/s)
α	Volume fraction (dimensionless)
β	Coefficient of thermal expansion (K^{-1})
γ	Porosity (dimensionless)
γ	Ratio of specific heats, c_p/c_v (dimensionless)
Δ	Change in variable, final − initial (for example, Δp, Δt, ΔH, ΔS, ΔT)

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

δ	Delta function (units vary)
ε	Emissivity (dimensionless)
ε	Lennard-Jones energy parameter (J/molecule)
ε	Turbulent dissipation rate (m^2/s^3, ft^2/s^3)
ε	Void fraction (dimensionless)
η	Effectiveness factor (dimensionless)
η', η''	Rate exponents for reactants, products (dimensionless)
θ_r	Radiation temperature (K)
λ	Molecular mean free path (m, nm, ft)
λ	Wavelength (m, nm, Å, ft)
μ	Dynamic viscosity (cP, Pa-s, lb_m/ft-s)
ν	Kinematic viscosity (m^2/s, ft^2/s)
ν', ν''	Stoichiometric coefficients for reactants, products (dimensionless)
ρ	Density (kg/m^3, lb_m/ft^3)
σ	Stefan-Boltzmann constant (5.67×10^{-8} $W/m^2\text{-}K^4$)
σ	Surface tension (kg/m, dyn/cm, lb_f/ft)
σ_s	Scattering coefficient (m^{-1})
$\bar{\bar{\tau}}$	Stress tensor (Pa, lb_f/ft^2)
τ	Shear stress (Pa, lb_f/ft^2)
τ	Time scale, for example, τ_c, τ_p, τ_c (s)
τ	Tortuosity, characteristic of pore structure (dimensionless)
ϕ	Equivalence ratio (dimensionless)
ϕ	Thiele modulus (dimensionless)
Ω	Angular velocity; Ω_{ij}, Mean rate of rotation tensor (s^{-1})
ω	Specific dissipation rate (s^{-1})
Ω,Ω'	Solid angle (degrees, radians, gradians)
Ω_D	Diffusion collision integral (dimensionless)

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information
of ANSYS, Inc. and its subsidiaries and affiliates.

Bibliography

[1] S. Wallin and A. Johansson. "A complete explicit algebraic Reynolds stress model for incompressible and compressible flows". *Journal of Fluid Mechanics*. 403. 89–132. 2000.

[2] *Fuel Cell Handbook (5th Edition)*. E G & G Services and Parsons Inc. and Science Applications International Corporation. 2000.

[3] Recommended Practice DNV-RP-C205. Environmental Conditions and Environmental Loads. Det Norske Veritas. October 2010.

[4] "Recommended Practice - RP O501 - Erosive Wear in Piping Systems". *Rev 4.2*. Det Norske Veritas. 2007.

[5] *User Manual for 1993 Version Pulverized Coal Gasification and Combustion 3-Dimensional (pcgc-3)*. Advanced Combustion Engineering Research Center, Brigham Young University. 1993.

[6] J. Abrahamson. "Collision Rates of Small Particles in a Vigorously Turbulent Fluid". *Chemical Engineering Science*. 30. 1371–1379. 1975.

[7] B. Abramzon and Sirignano. "Droplet Vaporization Model for Spray Combustion Calculations". *Int. J. Heat Mass Transfer*. 32 (9). 1605-1689. 1989.

[8] M. Agrawal, A. Bakker, and M. T. Prinkey. "Macroscopic Particle Model – Tracking Big Particles in CFD". *AIChE 2004 Annual Meeting*. Particle Technology Forum - Paper 268b. Austin, Texas, USA: 2004.

[9] M. Agrawal, K. Ogawa. "Drag force formulation in macroscopic particle model and its validation". *AIChE 2009 Annual Meeting*. Paper 163b. Nashville, TN, USA: 2009.

[10] T. Ahmad, S. L. Plee, and J. P. Myers. "Computation of Nitric Oxide and Soot Emissions from Turbulent Diffusion Flames". *J. of Engineering for Gas Turbines and Power*. 107. 48–53. 1985.

[11] F. Akao, K. Araki, S. Mori, and A. Moriyama. *Deformation Behaviors of a Liquid Droplet Impinging onto Hot Metal Surface*. 107. 737–743. Trans. Iron and Steel Institute of Japan. 1980.

[12] 1. A. Åkesjö, M. Gourdon, L. Vamling, F. Innings, and S. Sasic . *"Experimental and numerical study of heat transfer in a large–scale vertical falling film pilot unit". Int. J. of Heat and Mass Transfer*. 125. 53–65. 2018.

[13] Shah Shahood Alam et al. "Single and multicomponent droplet models for spray applications". *American J. Energy Eng.*. Vol. 2. No. 5. 108–126. 1960.

[14] B. J. Alder and T. E. Wainwright. "Studies in Molecular Dynamics II: Behaviour of a Small Number of Elastic Spheres". *J. Chem. Phys*. 33. 1439. 1960.

[15] S. Alfaki, W. Tabakoff. "Erosion Study of Radial Flow Compressor With Splitters". *Transactions of the ASME*. 109. 62-69. January 1987.

[16] A. A. Amsden. "KIVA-3: A KIVA Program with Block-Structured Mesh for Complex Geometries". *Technical Report LA-12503-MS, UC-361*. Los Alamos National Laboratory, Los Alamos, New Mexico. March 1993.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

[17] M. Andersson, X. Lu, J. Yuan, and B. Sundén. "Analysis of Microscopic Anode Structure Effects on an Anode-Supported SOFC Including Knudsen Diffusion". *ECS Transactions*. 35 (1). 1799–1809. 2011.

[18] T. B. Anderson and R. Jackson. "A Fluid Mechanical Description of Fluidized Beds". *I & EC Fundam*. 6. 527–534. 1967.

[19] W. Anderson and D. L. Bonhus. "An Implicit Upwind Algorithm for Computing Turbulent Flows on Unstructured Grids". *Computers Fluids*. 23(1). 1–21. 1994.

[20] S. P. Antal, R. T. Lahey, and J. E. Flaherty. "Analysis of phase distribution in fully developed laminar bubbly two-phase flow". *International Journal of Multiphase Flow*. 17. 5. 635–652. 1991.

[21] A. Antifora, M. Sala, A. Perera, and L. Vigevano. "NOx Emissions in Combustion Systems of Coal Fired Furnaces with a Reducing Environment: Predictions and Measurements". *Fourth International Conference on Technologies and Combustion for a Clean Environment*. Lisbon, Portugal. 1997.

[22] J. Appel, H. Bockhorn, and M. Frenklach. "Kinetic Modeling of Soot Formation with Detailed Chemistry and Physics: Laminar Premixed Flames of C2 Hydrocarbons". *Combustion and Flame*. 121. 122 – 136. 2000.

[23] D.D. Apsley and M.A. Leschziner. "A new low-reynolds-number nonlinear two-equation turbulence model for complex flows". *International Journal of Heat and Fluid Flow*. 19. 209–222. 1998.

[24] S. V. Apte, M. Gorokhovski, and P. Moin. "LES of Atomizing Spray with Stochastic Modeling of Secondary Breakup". *International Journal of Multiphase Flow*. 29. 1503 –1522. 2003.

[25] S. V. Apte, K. Mahesh, and T. Lundgren. "Accounting for finite-size effects in simulations of dispersed particle-laden flows". *International Journal of Multiphase Flow*. 34(3). 260–271. 2008.

[26] D. Aquaro and E. Fontani. "Erosion of ductile and brittle materials: comparison between theoretical and numerical models and experimental data". *AIMETA International Tribology Conference*. 20-22. Aquila, Italy. September 2000.

[27] D. Aquaro and E. Fontani. "Erosion of ductile and brittle materials". *E. Meccanica*. 36. 651-661. Aquila, Italy. 2001.

[28] S. Armsfield and R. Street. "The Fractional-Step Method for the Navier-Stokes Equations on Staggered Grids: Accuracy of Three Variations". *Journal of Computational Physics*. 153. 660–665. 1999.

[29] D. Arnal, M. Habiballah, and E. Coustols. "Théorie de l'instabilité laminaire et critères de transition en écoulement bi et tridimensionnel". *La Recherche aérospatiale*. 2. 125–143. 1984.

[30] M. J. Augenstein, A. M. Tenenbaum, Y. Langsam. *Data Structures Using C*. Prentice Hall. 1990.

[31] R.H. Aungier. *"Centrifugal Compressors: A strategy for Aerodynamic Design and Analysis"*. ASME Press, New York. 2000.

[32] B. Aupoix. *"Roughness Corrections for the k-ω Shear Stress Transport Model: Status and Proposals "* . Journal of Fluids Engineering. 137.

[33] B. Aupoix, P. R. Spalart. *"Extensions of the Spalart–Allmaras turbulence model to account for wall roughness "*. International Journal of Heat and Fluid Flow. 2003. 24. 454-462.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

[34] L. Austin, K. Shoji, V. Bhatia, V. Jindal, K. Savage, and R. Klimpel. “Some Results on the Description of Size Distribution as a Rate Process in Various Mills”. *Industrial Engineering and Chemical Process Design Devices*. 15(1). 187–196. 1976.

[35] F. Backmier, K. H. Eberius, and T. Just. *Comb. Sci. Tech*. 7. 77. 1973.

[36] S. Badzioch and P. G. W. Hawksley. "Kinetics of Thermal Decomposition of Pulverized Coal Particles". *Ind. Eng. Chem. Process Design and Development*. 9. 521–530. 1970.

[37] C. Bai and A. Gosman. "Development of Methodology for Spray Impingement Simulation". *"SAE Technical Paper 950283"*. 1995.

[38] J. Baldyga. "Turbulent mixer model with application to homogeneous instantaneous chemical reactions". *Chem. Eng. Sci*. 44. 1175–1182. 1989.

[39] J. Baldyga and W. Orciuch. “Barium Sulfate Precipitation in a Pipe – An Experimental Study and CFD Modeling”. *Chemical Engineering Science*. 56. 2435–2444. 2001.

[40] R. S. Barlow, G. J. Fiechtner, C. D. Carter, and J. Y. Chen. "Experiments on the Scalar Structure of Turbulent $CO/H_2/N_2$ Jet Flames". *Combustion and Flame*. 120. 549–569. 2000.

[41] F. J. Barnes, J. H. Bromly, T. J. Edwards, and R. Madngezewsky. "NOx Emissions from Radiant Gas Burners". *Journal of the Institute of Energy*. 155. 184–188. 1988.

[42] J. C. Barrett and C. F. Clement. "Growth rates for liquid drops". *J. Aerosol. Sci.*. 19(2). 223-242. 1988.

[43] Y. Bartosiewicz, J. Lavieville, J.M. Seynhaeve. "A first assessment of the NEPTUNE_CFD code: instabilities in a stratified flow comparison between the VOF method and a two-field approach". *International Journal of Heat and Fluid Flow*. 29. 2. 2008.

[44] T. J. Barth and D. Jespersen. "The design and application of upwind schemes on unstructured meshes". *Technical Report AIAA-89-0366*. AIAA 27th Aerospace Sciences Meeting, Reno, Nevada. 1989.

[45] H. Barths, C. Antoni, and N. Peters. "Three-Dimensional Simulation of Pollutant Formation in a DI-Diesel Engine Using Multiple Interactive Flamelets". *SAE Paper*. SAE. 1998.

[46] H. Barths, N. Peters, N. Brehm, A. Mack M. Pfitzner, and V. Smiljanowski. "Simulation of pollutant formation in a gas turbine combustor using unsteady flamelets". *27th Symp. (Int'l.) on Combustion*. The Combustion Institute. 1841–1847. 1998.

[47] G. K. Batchelor. *An Introduction to Fluid Dynamics*. Cambridge Univ.Press. Cambridge, England. 1967.

[48] D. L. Baulch, D. D. Drysdall, D. G. Horne, and A. C. Lloyd. *Evaluated Kinetic Data for High Temperature Reactions*. 1,2,3. Butterworth. 1973.

[49] D. L. Baulch et al. "Evaluated Kinetic Data for Combustion Modelling". *J. Physical and Chemical Reference Data*. 21(3). 1992.

[50] M. M. Baum and P. J. Street. "Predicting the Combustion Behavior of Coal Particles". *Combust. Sci. Tech*. 3(5). 231–243. 1971.

[51] K. J. Baumeister and F. F. Simon. "Leidenfrost Temperature—Its Correlation for Liquid Metals, Cryogens, Hydrocarbons, and Water". *J. Heat Transfer*. 95(2). 166–173. May 1973.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

[52] L. L. Baxter. "Turbulent Transport of Particles". *PhD Thesis*. Brigham Young University, Provo, Utah. 1989.

[53] L. L. Baxter and P. J. Smith. "Turbulent Dispersion of Particles: The STP Model". *Energy & Fuels*. 7. 852–859. 1993.

[54] J. C. Beale and R. D. Reitz. "Modeling Spray Atomization with the Kelvin-Helmholtz/Rayleigh-Taylor Hybrid Model". *Atomization and Sprays*. 9. 623–650. 1999.

[55] W. Bechara, C. Bailly, P. Lafon, and S. Candel. "Stochastic Approach to Noise Modeling for Free Turbulent Flows". *AIAA Journal*. 32. 3. 1994.

[56] M. Behnia, S. Parneix, Y. Shabany, and P. A. Durbin. "Numerical Study of Turbulent Heat Transfer in Confined and Unconfined Impinging Jets". *International Journal of Heat and Fluid Flow*. 20. 1–9. 1999.

[57] F. Bhaktar, M. Piran. "Thermodynamic Properties of Supercooled Steam". *Int. J. Heat and Fluid Flow*. 1(2). 53–62. 1979.

[58] F. Bhaktar, K Zidi. "Nuleation Phenomena in Flowing High-Pressure Steam". *Part 2: Theoretical Analysis, Proc. Instn. Mech. Engrs., Part A: Journal of Power and Energy*. 204. 233–242.

[59] M. Asghar Bhatti. *Advanced Topics in Finite Element Analysis of Structures: With Mathematica and MATLAB Computations*. John Wiley and Sons, New York. 2006.

[60] R. W. Bilger and R. E. Beck. *15th Symp. (Int'l.) on Combustion*. The Combustion Institute. 541. 1975.

[61] R. W. Bilger, M. B. Esler, and S. H. Starner. "On Reduced Mechanisms for Methane-Air Combustion". *Lecture Notes in Physics*. Springer-Verlag. 384. page 86. 1991.

[62] B. Binniger, M. Chan, G. Paczkko, and M. Herrmann. "Numerical Simulation of Turbulent Partially Premixed Hydrogen Flames with the Flamelet Model". Technical reportAdvanced Combustion Gmbh. Internal Report. 1998.

[63] G. A. Bird. *Molecular Gas Dynamics and the direct Simulation of Gas Flows*. Oxford University Press, New York. 1994.

[64] G. Blanquart, H. Pitsch. "A Joint Volume-Surface-Hydrogen Multi-Variate Model for Soot Formation, in Combustion Generated Fine Carbonaceous Particles". Karlsruhe University Press. 439-466. 2009.

[65] F. Birkhold. "Selektive katalytische Reduktion von Stickoxiden in Kraftfahrzeugen: Untersuchung der Einspritzung von Harnstoffwasserlösung". Dissertation. University of Karlsruhe. 541. 2007.

[66] F. Birkhold, U. Meingast, P. Wassermann, and O. Deutschmann. "Analysis of the Injection of Urea-Water-Solution for Automotive SCR DeNOx-Systems: Modeling of Two-Phase Flow and Spray/Wall-Interaction". *SAE Technical Paper 2006-01-0643*. University of Karlsruhe. 2006.

[67] F. Birkhold, U. Meingast, P. Wassermann, and O. Deutschmann. "Modeling and simulation of the injection of urea-water-solution for automotive SCR DeNOx-systems". *Applied Catalysis B: Environmental*. 70(1). 119-127. 2007.

[68] J. Blauvens, B. Smets, and J. Peters. *In 16th Symp.(Int'l.) on Combustion*. The Combustion Institute. 1977.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

[69] Blint. *Principles of Combustion*. Wiley-Interscience. New York1988.

[70] P. Boudier and et al. "A model for Turbulent Flame Ignition and Propagation in Spark Ignition Engines". *In Twenty Fourth Symposium (international) on Combustion, Sydney, Australia*. 1992.

[71] R. M. Bowen. "Theory of Mixtures". In A. C. Eringen, editor*Continuum Physics*. Academic Press, New York. 1–127. 1976.

[72] C. T. Bowman. "Chemistry of Gaseous Pollutant Formation and Destruction". In W. Bartok and A. F. Sarofim, editors*Fossil Fuel Combustion*. J. Wiley and Sons, Canada. 1991.

[73] R. K. Boyd and J. H. Kent. "Three-dimensional furnace computer modeling". *In 21st Symp. (Int'l.) on Combustion*. The Combustion Institute. 265–274. 1986.

[74] J. U. Brackbill, D. B. Kothe, and C. Zemach. "A Continuum Method for Modeling Surface Tension". *J. Comput. Phys*. 100. 335–354. 1992.

[75] A. Brandt. "Multi-level Adaptive Computations in Fluid Dynamics". *Technical Report AIAA-79-1455*. AIAA, Williamsburg, VA. 1979.

[76] K. Bray, P. Domingo and L. Vervisch. "Role of the Progress Variable in Models for Partially Premixed Turbulent Combustion". *Combustion and Flame*. 141. 431–437. 2005.

[77] K. N. Bray and N. Peters. "Laminar Flamelets in Turbulent Flames". In P. A. Libby and F. A. Williams, editors*Turbulent Reacting Flows*. Academic Press. 63–114. 1994.

[78] C. E. Brennen. *Cavitation and Bubble Dynamics*. Oxford University Press. 1995.

[79] K. S. Brentner and F. Farassat. "An Analytical Comparison of the Acoustic Analogy and Kirchhoff Formulations for Moving Surfaces". *AIAA Journal*. 36(8). 1998.

[80] M. Brocchini, D.H. Peregrine. "The dynamics of strong turbulence at free sur-faces, part 1: description". *J. Fluid Mech.*. 449. 225–254. 2001.

[81] S. J. Brookes and J. B. Moss. "Prediction of Soot and Thermal Radiation in Confined Turbulent Jet Diffusion Flames". *Combustion and Flame*. 116. 486–503. 1999.

[82] J. Brouwer, M. P. Heap, D. W. Pershing, and P. J. Smith. "A Model for Prediction of Selective Non-Catalytic Reduction of Nitrogen Oxides by Ammonia, Urea, and Cyanuric Acid with Mixing Limitations in the Presence of CO". *In 26th Symposium (Int'l) on Combustion*. The Combustion Institute. 1996.

[83] A. L. Brown. "Modeling Soot in Pulverized Coal Flames". *MSc thesis*. Brigham Young University, Utah, USA. 1997.

[84] A. Brucato, F. Grisafi, and G. Montante. "Particle drag coefficients in turbulent fluids". *Chemical Engineering Science*. 53(18). 3295–3314. 1998.

[85] S. Brunauer. *The Absorption of Gases and Vapors*. Princeton University Press, Princeton, NJ. 1943.

[86] M. Budakli . "Hydrodynamics and Heat Transfer in Gas–Driven Liquid Film Flow". *PhD Thesis*. Institute of Technical Thermodynamics of the Technische Universitat Darmstadt, Germany. 2015.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

[87] Trong T. Bui. "A Parallel Finite-Volume Algorithm for Large-Eddy Simulation of Turbulent Flows". *Technical Memorandum NASA/TM-1999-206570*. 1999.

[88] A. D. B. Burns, Th. Frank, I. Hamill, and J.-M. Shi. The Favre Averaged Drag Model for Turbulent Dispersion in Eulerian Multi-Phase Flows. *Fifth International Conference on Multiphase Flow, ICMF-2004, Yokohama, Japan*. 2004.

[89] D. Butterworth. "The correlation of cross flow pressure drop data by means of a permeability concept". *UKAEA Report AERE–R9435*. 1979.

[90] L. Cai and R. E. White. "Reduction of Model Order Based on Proper Orthogonal Decomposition for Lithium-Ion Battery Simulations". *J. of Electrochemical Soc.*. 156(3). A154-A161. 2009.

[91] S. Candel and T. Poinsot. "Flame stretch and the balance equation for the flame area". *Combustion Science and Technology*. 70. 1–15. 1990.

[92] R. Cao and S. B. Pope. "Numerical Integration of Stochastic Differential Equations: Weak Second-Order Mid-Point Scheme for Application in the Composition PDF Method". *Journal of Computational Physics*. 185(1). 194–212. 2003.

[93] N. F. Carnahan and K. E. Starling. "Equations of State for Non-Attracting Rigid Spheres". *J. Chem. Phys*. 51. 635–636. 1969.

[94] M. G. Carvalho, T. Farias, and P. Fontes. "Predicting Radiative Heat Transfer in Absorbing, Emitting, and Scattering Media Using the Discrete Transfer Method". In W. A. Fiveland et al., editor*Fundamentals of Radiation Heat Transfer*. 160. ASME HTD. pages 17–26. 1991.

[95] J. R. Cash and A. H. Karp. "A variable order Runge-Kutta method for initial value problems with rapidly varying right-hand sides". *ACM Transactions on Mathematical Software*. 16. 201–222. 1990.

[96] G.V. Candler, and R.W. MacCormack. "Computation of Weakly Ionized Hypersonic Flows in Thermochemical Nonequilibrium". *Journal of Thermophysics and Heat Transfer* . 5. 3. 266–273. 1991.

[97] Cantera. http://sourceforge.net/projects/cantera/ . 2012.

[98] T. Cebeci and P. Bradshaw. *Momentum Transfer in Boundary Layers*. Hemisphere Publishing Corporation, New York. 1977.

[99] S. Chapman and T. G. Cowling. *The Mathematical Theory of Non-Uniform Gases*. 3rd edition. Cambridge University Press, Cambridge, England. 1990.

[100] F. Charlette, C. Meneveau, and D. Veynante. "A Power-Law Flame Wrinkling Model for LES of Premixed Turbulent Combustion, Part I: Non-Dynamic Formulation and Initial Tests". *Combustion and Flame*. 131. 1–2. 159–180. 2002.

[101] S. Charpenay, M. A. Serio, and P. R. Solomon. *In 24th Symp. (Int'l.) on Combustion*. 1189–1197. The Combustion Institute. 1992.

[102] *CHEMKIN-PRO 15112*. Reaction Design. San Diego: 2012.

[103] H. C. Chen and V. C. Patel. "Near-Wall Turbulence Models for Complex Flows Including Separation". *AIAA Journal*. 26(6). 641–648. 1988.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

[104] J. C. Chen. "Correlation for Boiling Heat Transfer to Saturated Fluids in Convective Flow". *Industrial & Engineering Chemistry Process Design and Development*. 5. 322–329. 1966.

[105] M. Chen and G. A. Rincon-Mora. "Accurate Electrical Battery Model Capable of Predicting Runtime and I-V Performance". *IEEE Trans. On Energy Conversion*. Vol. 21. No.2. A154-A161. June 2006.

[106] P. Cheng. "Two-Dimensional Radiating Gas Flow by a Moment Method". *AIAA Journal*. 2. 1662–1664. 1964.

[107] R. V. Chima and M. S. Liou. "Comparison of the AUSM+ and H-CUSP Schemes for Turbomachinery Applications". *NASA TM-2003-212457*. 2003.

[108] A. J. Chorin. "Numerical solution of navier-stokes equations". *Mathematics of Computation*. 22. 745–762. 1968.

[109] B. H. Chowdhury. "Emission Control Alternatives for Electric Utility Power Plants". *Energy Sources*. 18(4). 393–406. October 1996.

[110] E. H. Chui and G. D. Raithby. "Computation of Radiant Heat Transfer on a Non-Orthogonal Mesh Using the Finite-Volume Method". *Numerical Heat Transfer, Part B*. 23. 269–288. 1993.

[111] Clift, Grace, and Weber. "Bubbles, Drops, and Particles". *Technical Report*. Academic Press. 1978.

[112] P. J. Coelho and N. Peters. "Unsteady Modelling of a Piloted Methane/Air Jet Flame Based on the Eulerian Particle Flamelet Model". *Combustion and Flame*. 122. 444–465. 2001.

[113] M. F. Cohen and D. P. Greenberg. "The hemi-cube: a radiosity solution for complex environments". *ACM SIGGRAPH Computer Graphics*. 19(3). 31–40. 1985.

[114] D. Cokljat, V. A. Ivanov, F. J. Sarasola, and S. A. Vasquez. "Multiphase k-epsilon Models for Unstructured Meshes". *In ASME 2000 Fluids Engineering Division Summer Meeting*. Boston, MA, USA. 2000.

[115] D. Cokljat, M. Slack, and S. A. Vasquez. "Reynolds-Stress Model for Eulerian Multiphase". In Y. Nagano K. Hanjalic and M. J. Tummers, editors. *Proceedings of the 4th International Symposium on Turbulence Heat and Mass Transfer*. Begell House, Inc. 1047–1054. 2003.

[116] R. Cole. "A Photographic Study of Pool Boiling in the Region of the Critical Heat Flux". *AIChE J*. 6. 533–542. 1960.

[117] H. W. Coleman and B. K. Hodge and R. P. Taylor. "A Re-Evaluation of Schlichting's Surface Roughness Experiment". *Journal of Fluids Engineering*. 106. 1984.

[118] O. Colin, C. Angelberger, and A. Benkenida. "3D Modeling of Mixing, Ignition and Combustion Phenomena in Highly Stratified Gasoline Engines". *Oil & Gas Science and Technology*. Rev IFP. 58. 1. 47–62. 2003.

[119] O. Colin, F. Ducros, D. Venante, T. Poinsot . "A thickened flame model for large eddy simulations of turbulent premixed combustion". *Physics of Fluids*. 12. 7. 1843–1863. 2000.

[120] P. T. Coman, E.C. Darcy, C.T. Veje, and R.E. White. "Modelling Li-Ion Cell Thermal Runaway Triggered by an Internal Short Circuit Device Using an Efficiency Factor and Arrhenius Formulations". *Journal of the Electrochemical Society*. 164. 4. A587–A593. 2017.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

[121] A. Coppalle and P. Vervisch. "The Total Emissivities of High-Temperature Flames". *Combustion and Flame*. 49. 101–108. 1983.

[122] S. M. Correa. "A Review of NOx Formation Under Gas-Turbine Combustion Conditions". *Combustion Science and Technology*. 87. 329–362. 1992.

[123] C. A. Coulaloglou and L. L. Tavlarides. “Description of Interaction Processes in Agitated Liquid-Liquid Dispersions”. *Chemical Engineering Science*. 32. 1289–1297. 1977.

[124] C. Crowe, M. Sommerfield, and Yutaka Tsuji. *Multiphase Flows with Droplets and Particles*. CRC Press. 1998.

[125] G. T. Csanady. "Turbulent Diffusion of Heavy Particles in the Atmosphere". *J. Atmos. Science*. 20. 201–208. 1963.

[126] P. A. Cundall and O. D. L. Strack. "A Discrete Numerical Model for Granular Assemblies". *Geotechnique*. 29. 47–65. 1979.

[127] Yang H.Q., Shinghal A.K., Megahed M., . "The Full Cavitation Model". *VKI, Lecture Series 2005-04, Industrial Two-Phase Flow CFD*. May 23-27, 2005.

[128] N. Curle. "The Influence of Solid Boundaries upon Aerodynamic Sound". *Proceedings of the Royal Society of London*. Series A, Mathematical and Physical Sciences, 231. 505–514. 1955.

[129] J. Dacles-Mariani, G. G. Zilliac, J. S. Chow, and P. Bradshaw. "Numerical/Experimental Study of a Wingtip Vortex in the Near Field". *AIAA Journal*. 33(9). 1561–1568. 1995.

[130] D. Dajnak and F. C. Lockwood. "Modelling of Toxic Heavy Metal Mercury Partitioning from Pulverized Fuel Combustion". *IFRF Combustion Journal*.

[131] J. M. Dalla Valle. *Micromeritics*. Pitman, London. 1948.

[132] B. J. Daly and F. H. Harlow. "Transport Equations in Turbulence". *Phys. Fluids*. 13. 2634–2649. 1970.

[133] S. Das, H. Punekar. "On Development of a Semi-Mechanistic Wall Boiling Model". *Journal of Heat Transfer*. 138. 6. 061501-1–10. 2005. February 2016.

[134] J. F. Daunenhofer and J. R. Baron. "Grid Adaption for the 2D Euler Equations". *Technical Report AIAA-85-0484*. American Institute of Aeronautics and Astronautics. 1985.

[135] Deendarlianto; T. Höhne, D. Lucas, C. Vallée, G. Montoya. "CFD studies on the phenomena around counter-current flow limitations of gas/liquid two-phase flow in a model of a PWR hot leg". *Nuclear Eng. and Des.*. 241. 12. 5138–5148. 2011.

[136] V. H. Del Valle and D. B. R. Kenning. "Subcooled flow boiling at high heat flux". *International Journal of Heat and Mass Transfer*. 28(10). 1907–1920. 1985.

[137] S. C. R. Dennis, S. N. Singh, D. B. Ingham. "The steady flow due to a rotating sphere at low and moderate Reynolds numbers". *J. Fluid Mech.*. 101. 257-279. 1980.

[138] M. K. Denison and B. W. Webb. "A Spectral Line-Based Weighted-Sum-of-Gray-Gases Model for Arbitrary RTE Solvers". *J. Heat Transfer*. 115. 1002–1012. 1993.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

[139] G. G. DeSoete. "Overall Reaction Rates of NO and N_2 Formation from Fuel Nitrogen". *In 15th Symp. (Int'l.) on Combustion*. The Combustion Institute. 1093–1102. 1975.

[140] H. Dette and W. J. Studden. *The Theory of Canonical Moments with Applications in Statistics, Probability, and Analysis*. John Wiley & Sons. New York, NY1997.

[141] R. B. Diemer and J. H. Olson. "A Moment Methodology for Coagulation and Breakage Problems Part 3—Generalized Daughter Distribution Functions". *Chemical Engineering Science*. 57(19). 4187–4198. 2002.

[142] J. Ding and D. Gidaspow. "A Bubbling Fluidization Model Using Kinetic Theory of Granular Flow". *AIChE J*. 36(4). 523–538. 1990.

[143] G. Dixon-Lewis. "Structure of Laminar Flames". *In 23rd Symp. (Int'l.) on Combustion*. The Combustion Institute. 305–324. 1990.

[144] N. Dombrowski and P. C. Hooper. "The effect of ambient density or drop formation in sprays". *Chemical Engineering Science*. 17. 291–305. 1962.

[145] N. Dombrowski and W. R. Johns. "The Aerodynamic Instability and Disintegration of Viscous Liquid Sheets". *Chemical Engineering Science*. 18. 203. 1963.

[146] C. Dopazo and E. E. O'Brien. "Functional Formulation of Nonisothermal Turbulent Reactive Flows". *Phys. Fluids*. 17. 1968. 1975.

[147] A. M. Douaud and P. Eyzat. "Four-Octane-Number Method for Predicting the Anti-Knock Behavior of Fuels in Engines". *SAE Technical Paper*. v87 780080. SAE. 1978.

[148] M. Doyle, T. F. Fuller and J. Newman. "Modeling of Galvanostatic Charge and Discharge of the Lithium/Polymer/Insertion Cell". *J. of Electrochemical Soc.*. Vol 140, No. 6. 1526-1533. 1993.

[149] M. C. Drake and R. J. Blint. "Relative Importance of Nitrogen Oxide Formation Mechanisms in Laminar Opposed-Flow Diffusion Flames". *Combustion and Flame*. 83. 185–203. 1991.

[150] M. C. Drake, S. M. Correa, R. W. Pitz, W. Shyy, and C. P. Fenimore. "Superequilibrium and Thermal Nitric Oxide Formation in Turbulent Diffusion Flames". *Combustion and Flame*. 69. 347–365. 1987.

[151] M. C. Drake, R. W. Pitz, M. Lapp, C. P. Fenimore, R. P. Lucht, D. W. Sweeney, and N. M. Laurendeau. *In 20th Symp. (Int'l.) on Combustion*. The Combustion Institute. 327. 1984.

[152] D. A. Drew and R. T. Lahey. *In Particulate Two-Phase Flow*. Butterworth-Heinemann. Boston, MA509–566. 1993.

[153] J. K. Dukowwicz and A. S. Dvinsky. "Approximate Factorization as a High-Order Splitting for the Implicit Incompressible Flow Equations". *Journal of Computational Physics*. 102. 336–347. 1992.

[154] L. Durand and W. Polifke. "Implementation of the Thickened Flame Model for Large Eddy Simulation of Turbulent Premixed Combustion in a Commercial Solver". *ASME Turbo Expo 2007: Power for Land, Sea, and Air*. GT2007-28188. 869–878. 2007.

[155] P. A. Durbin. "Separated Flow Computations with the k- ε- v^2 Model". *AIAA Journal*. 33(4). 659–664. 1995.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

[156] Brandon M. Eaton. "One-dimensional, Transient Model of Heat, Mass, and Charge Transfer in a Proton Exchange Membrane". *M.S. Thesis*. 2001.

[157] E. R. G. Eckert and R. M. Drake. *Analysis of Heat and Mass Transfer*. McGraw-Hill Co. 1972.

[158] D. K. Edwards and R. Matavosian. "Scaling Rules for Total Absorptivity and Emissivity of Gases". *J. Heat Transfer*. 106. 684–689. 1984.

[159] J. K. Edwards, B. S. McLaury, and S. A. Shirazi. "Supplementing a CFD Code with Erosion Prediction Capabilities". *In Proceedings of ASME FEDSM'98: ASME 1998 Fluids Engineering Division Summer Meeting, Washington DC*. June 1998.

[160] J. K. Edwards, B. S. McLaury, and S. A. Shirazi. "Evaluation of Alternative Pipe Bend Fittings in Erosive Service". *In Proceedings of ASME FEDSM'00: ASME 2000 Fluids Engineering Division Summer Meeting, Boston, MA*. June 2000.

[161] Y. Egorov. "Contact Condensation in Stratified Steam-Water Flow". *EVOL-ECORDA-D 07*. 2004.

[162] Y. Egorov, F. Menter. "Contact Condensation in Stratified Steam–Water Flow". *EVOL-ECORA-D 07*. 2004.

[163] S. E. Elgobashi and T. W. Abou-Arab. "A Two-Equation Turbulence Model for Two-Phase Flows". *Phys. Fluids*. 26(4). 931–938. 1983.

[164] S. Ergun. "Fluid Flow through Packed Columns". *Chem. Eng. Prog*. 48(2). 89–94. 1952.

[165] G. Erlebacher, M. Y. Hussaini, C. G. Speziale, and T. A. Zang. "Toward the Large-Eddy Simulation of Compressible Turbulent Flows". *J. Fluid Mech*. 238. 155–185. 1992.

[166] M.J. Evans, C. Petre, P.R. Medwell, and A. Parente. "Generalisation of the eddy-dissipation concept for jet flames with low turbulence and low Damköhler number". *Proceedings of the Combustion Institute*. 37. 4497–4505. 2019.

[167] J. Ewald. "A Level Set Based Flamelet Model for the Prediction of Combustion in Homogeneous Charge and Direct Injection Spark Ignition Engines". *PhD Thesis*. Aachen University. 2006.

[168] R. Ewert and W. Schroeder. "Acoustic perturbation equations based on flow decomposition via source filtering". *Journal of Computational Physics*. Volume 188, Issue 2. 365–398. 2003.

[169] R. Fan, D. L. Marchisio and R. O. Fox. Application of the Direct Quadrature Method of Moments to Poly-disperse Gas-solid Fluidized Beds. *Power Technology*. 139. 7–20. 2004.

[170] R. F. Fedors and R. F. Landell. "An Empirical Method of Estimating the Void Fraction in Mixtures of Uniform Particles of Different Size". *Powder Technology*. 23. 225–231.

[171] C. P. Fenimore. "Formation of Nitric Oxide in Premixed Hydrocarbon Flames". *In 13th Symp.(Int'l.)on Combustion*. The Combustion Institute. 373. 1971.

[172] C. P. Fenimore. "Destruction of NO by NH_3 in Lean Burnt Gas". *Combustion and Flame*. 37. 245. 1980.

[173] C. P. Fenimore and G. W. Jones. "Oxidation of soot by hydroxyl radicals". *J. Phys. Chem*. 71. 593–597. 1967.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

[174] John D. Fenton. "A Fifth-Order Stokes Theory for Steady Waves". *Journal of Waterway, Port, Coastal and Ocean Engineering*. 111, No.2. March 1985.

[175] John D. Fenton. "The Cnoidal Theory of Water Waves". *Developments in Offshore Engineering*. J. B. Herbich, Houston. 1998.

[176] M. Ferrarotti, Z. Li, and A. Parente. "On the role of mixing models in the simulation of MILD combustion using finite-rate chemistry combustion models". *Proceedings of the Combustion Institute*. 37. 4. 4531–4538. 2019.

[177] J. L. Ferzieger and M. Peric. *Computational Methods for Fluid Dynamics*. Springer-Verlag, Heidelberg. 1996.

[178] J. E. Ffowcs-Williams and D. L. Hawkings. "Sound Generation by Turbulence and Surfaces in Arbitrary Motion". *Proc. Roy. Soc. London*. A264. 321–342. 1969.

[179] M. A. Field. "Rate of Combustion of Size-Graded Fractions of Char from a Low Rank Coal between 1200 K–2000 K". *Combustion and Flame*. 13. 237–252. 1969.

[180] I. Finnie. "Erosion of Surfaces by Solid Particles". *Wear*. 3. 87–103. 1960.

[181] W. A. Fiveland and A. S. Jamaluddin. "Three-Dimensional Spectral Radiative Heat Transfer Solutions by the Discrete Ordinates Method". *Heat Transfer Phenomena in Radiation, Combustion and Fires*. HTD Vol. 10643–48. 1989.

[182] T. H. Fletcher and D. R. Hardesty. "Compilation of Sandia coal devolatilization data: Milestone report". *Sandia Report SAND92-8209*. 1992.

[183] T. H. Fletcher and A. R. Kerstein. "Chemical percolation model for devolatilization: 3. Direct use of ^{13}C NMR data to predict effects of coal type". *Energy and Fuels*. 6. 414. 1992.

[184] T. H. Fletcher, A. R. Kerstein, R. J. Pugmire, and D. M. Grant. "Chemical percolation model for devolatilization: 2. Temperature and heating rate effects on product yields". *Energy and Fuels*. 4. 54. 1990.

[185] P. J. Flory. “Thermodynamics of High Polymer Solutions”. *Journal of Chemical Physics*. 10. 51–61. 1942.

[186] W. L. Flower, R. K. Hanson, and C. H. Kruger. *In 15th Symp. (Int'l.) on Combustion*. The Combustion Institute. 823. 1975.

[187] R. O. Fox. *Computational Models for Turbulent Reacting Flows*. Cambridge University Press, Cambridge, England. 2003.

[188] J. A. Foucart, *Private Communication* .

[189] Th. Frank. "Parallele Algorithmen für die numerische Simulation dreidimensionaler, disperser Mehrphasenströmungen und deren Anwendung in der Verfahrenstechnick". *Chemnitz, Techn. Univ., Habil-Schr*. Shaker Verlag. 2002.

[190] Th. Frank, J. M. Shi, and A. D. Burns. "Validation of Eulerian Multiphase Flow Models for Nuclear Safety Applications". *Third International Symposium on Two-Phase Flow Modeling and Experimentation, Pisa, Italy*. Sept. 22–24, 2004.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

[191] Th. Frank, P. J. Zwart, E. Krepper, H.-M. Prasser, and D. Lucas. "Validation of CFD models for mono- and polydisperse air-water two-phase flows in pipes". *J. Nuclear Engineering & Design*. 238. 647–659. March 2008.

[192] M. Frenklach. "Method of moments with interpolative closure". *Chemical Engineering Science*. 57. 2229-2239. 2002.

[193] M. A. Friedrich, H. Lan, J. L. Wegner, J. A. Drallmeier, and B. F. Armaly . "A Separation Criterion with Experimental Validation for Shear-Driven Films in Separated Flows". *Journal of Fluids Engineering*. 130. May 2008.

[194] S. Fu, B. E. Launder, and M. A. Leschziner. "Modeling Strongly Swirling Recirculating Jet Flow with Reynolds-Stress Transport Closures". *In Sixth Symposium on Turbulent Shear Flows*. Toulouse, France. 1987.

[195] B. Gampert. Grenzschichttheoretische Probleme des aerodynamischen Schmelzspinnprozesses. *PhD Thesis from the Technical University of Berlin*. Berlin, Germany 1973.

[196] F. Garcia-Ochoa and E. Gomez. "Bioreactor scale-up and oxygen transfer rate in microbial processes: An overview". *Biotechnology Advances*. 27(2). 153–176. 2009.

[197] J. Garside and M. R. Al-Dibouni. "Velocity-Voidage Relationships for Fluidization and Sedimentation". *I & EC Process Des. Dev*. 16. 206–214. 1977.

[198] J. A. Gasper, N. C. Widmer, J. A. Cole, and W. R. Seeker. "Study of Mercury Speciation in a Simulated Municipal Waste Incinerator Flue". *In Proceedings of 1997 International Conference on Incineration and Thermal Treatment Technologies*. Oakland, California. 1997.

[199] M. Germano, U. Piomelli, P. Moin, and W. H. Cabot. "Dynamic Subgrid-Scale Eddy Viscosity Model". *In Summer Workshop*. Center for Turbulence Research, Stanford, CA. 1996.

[200] T. Gessner. "Dynamic Mesh Adaption for Supersonic Combustion Waves Modeled with Detailed Reaction Mechanisms". *PhD thesis*. University of Freiburg, Freiburg, Germany. 2001.

[201] M. Ghadiri and Z. Zhang. “Impact Attrition of Particulate Solids Part 1. A Theoretical Model of Chipping”. *Chemical Engineering Science*. 57. 3659–3669. 2002.

[202] L. G. Gibilaro, R. Di Felice and S. P. Waldram. "Generalized Friction Factor and Drag Coefficient Correlations for Fluid-particle Interactions". *Chem. Eng. Sci*. 40(10). 1817-1823. 1985.

[203] M. M. Gibson and B. E. Launder. "Ground Effects on Pressure Fluctuations in the Atmospheric Boundary Layer". *J. Fluid Mech*. 86. 491–511. 1978.

[204] D. Gidaspow. *Multiphase Flow and Fluidization*. Academic Press, Boston, MA. 1994.

[205] D. Gidaspow, R. Bezburuah, and J. Ding. "Hydrodynamics of Circulating Fluidized Beds, Kinetic Theory Approach". *In Fluidization VII, Proceedings of the 7th Engineering Foundation Conference on Fluidization*. 75–82. 1992.

[206] R. G. Gilbert, K. Luther, and J. Troe. *Ber. Bunsenges. Phys. Chem*. 87. 1983.

[207] P. Glarborg, J. E. Johnsson, and K. Dam-Johansen. "Kinetics of Homogeneous Nitrous Oxide Decomposition”. *Combustion and Flame*. ,99. 523–532. 1994.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

[208] H. M. Glaz J. B. Bell, P. Colella. "Second-Order Projection Method for the Incompressible Navier-Stokes Equations". *Journal of Computational Physics*. 85. 257. 1989.

[209] H. M. Glaz J. B. Bell, P. Colella. "An Analysis of the Fractional-Step Method". *Journal of Computational Physics*. 108. 51–58. 1993.

[210] A. Gnanavelu, N. Kapur, A. Neville, J. F. Flores, N. Ghorbani. "A numerical investigation of a geometry independent integrated method to predict erosion rates in slurry erosion". *J. Wear*. 271. 712-719. 2011.

[211] P. A. Gnoffo, R. N. Gupta, and J. L. Shinn. "Conservation Equations and Physical Models for Hypersonic Air Flows in Thermal and Chemical Nonequilibrium". *NASA*. TP-2867. 1989.

[212] G. Goldin, A. Colclasure, A. Wiedemann, and R. Kee. "Three-dimensional Particle-resolved Models of Li-ion Batteries to Assist the Evaluation of Empirical Parameters in One-Dimensional Models". *Electrochemica Acta*. 64. 118–129. 2012.

[213] R. G. Gordon. "Error Bounds in Equilibrium Statistical Mechanics". *Journal of Mathematical Physics*. 56. 655–633. 1968.

[214] H. Görgün, M. Arcak, and F Barbir. "An algorithm for estimation of membrane water content in PEM fuel cells". *Journal of power sources* . 157. 1. 389–394. 2006.

[215] A. D. Gosman and E. Ioannides. "Aspects of computer simulation of liquid-fuelled combustors". *J. Energy*. 7(6). 482–490. 1983.

[216] J. Gottgens, F. Mauss, and N. Peters. "Analytic Approximations of Burning Velocities and Flame Thicknesses of Lean Hydrogen, Methane, Ethylene, Ethane, Acetylene and Propane Flames". *In Twenty-Fourth Symposium (Int.) on Combustion, Pittsburgh*. 29–135. 1992.

[217] I. R. Gran and B. F. Magnussen. "A numerical study of a bluff-body stabilized diffusion flame. part 2. influence of combustion modeling and finite-rate chemistry". *Combustion Science and Technology*. 119. 191. 1996.

[218] D. M. Grant, R. J. Pugmire, T. H. Fletcher, and A. R. Kerstein. "Chemical percolation model of coal devolatilization using percolation lattice statistics". *Energy and Fuels*. 3. 175. 1989.

[219] M. S. Gritskevich, A. V. Garbaruk, J. Schutze, F. R. Menter. "Development of DDES and IDDES Formulations for the k-ω Shear Stress Transport Model". *Flow, Turbulence and Combustion*. 88(3). 431–449. 2012.

[220] H. Gu. "Mathematical Analysis of a Zn/NiOOH Cell". *J. Electrochemical Soc.*. Princeton, NJ. 1459-1. 464. July 1983.

[221] W. Gumprich, M. Chrigui, M. Braun, A. Sadiki. "Evaluation of collision models applied to varying dense particle jets flows". *7th International Conference on Multiphase Flow*. Tampa, FL: May 30 – June 4, 2010.

[222] D. J. Gunn. "Transfer of Heat or Mass to Particles in Fixed and Fluidized Beds". *Int. J. Heat Mass Transfer*. 21. 467–476. 1978.

[223] Hagesaether L., Jakobsen H.A.1, and Svendsen H. F. "A Model for Turbulent Binary Breakup of Dispersed Fluid Particles". *Chemical Engineering Science*. 57(16). 3251–3267. 2002.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

[224] W. W. Hagerty and J. F. Shea. "A Study of the Stability of Plane Fluid Sheets". *Journal of Applied Mechanics*. 22. 509. 1955.

[225] A. Haider and O. Levenspiel. "Drag Coefficient and Terminal Velocity of Spherical and Nonspherical Particles". *Powder Technology*. 58. 63–70. 1989.

[226] B. Hall. "An Experimental Study of Mercury Reactions in Combustion Flue Gases". *PhD thesis*. Chalmers University of Technology, Sweden. 1991.

[227] R. J. Hall, M. D. Smooke, and M. B. Colket. *Physical and Chemical Aspects of Combustion*. Gordon and Breach. 1997.

[228] M. P. Halstead, L. J. Kirsch, and C. P. Quinn. "Autoignition of Hydrocarbon Fuels at High Temperatures and Pressures – Fitting of a Mathematical Model". *Combustion and Flame*. 30. 45–60. 1977.

[229] Z. Han, S. Perrish, P. V. Farrell, and R. D. Reitz. "Modeling Atomization Processes of Pressure-Swirl Hollow-Cone Fuel Sprays". *Atomization and Sprays*. 7(6). 663–684. Nov.-Dec. 1997.

[230] G. Hand, M. Missaghi, M. Pourkashanian, and A. Williams. "Experimental Studies and Computer Modeling of Nitrogen Oxides in a Cylindrical Furnace". *In Proceedings of the Ninth Members Conference*. volume 2. IFRF Doc No K21/g/301989.

[231] S. Hänsch, D. Lucas, Höhne, E. Krepper, G. Montoya. "Application of a Multi-Field Concept to the Dambreak Case with an Obstacle". *The 15th International Topical Meeting on Nuclear Reactor Thermalhydraulics (NURETH-15)*. Pisa, Italy. May 2013.

[232] S. Hänsch, D. Lucas, Höhne, E. Krepper, G. Montoya. "Comparative Simulations of Free Surface Flow using VOF-Methods and a New Approach for Multi-Scale Interfacial Structures". *ASME 2013 Fluids Engineering Summer Meeting FEDSM2013*. Nevada, USA. July 2013.

[233] R. K. Hanson and S. Salimian. "Survey of Rate Constants in H/N/O Systems". In W. C. Gardiner, editor*Combustion Chemistry*. 361. 1984.

[234] H. O. Hardenburg and F. W. Hase. "An Empirical Formula for Computing the Pressure Rise Delay of a Fuel from its Cetane Number and from the Relevant Parameters of Direct Injection Diesel Engines". *SAE Technical Paper 790493*. SAE. 1979.

[235] K. Haugen, O. Kvernvold, A. Ronald, and R. Sandberg. "Sand Erosion of Wear-resistant Materials: Erosion in Choke Valves.Wear". 186-187. 179–188. 1995.

[236] A. Hellsten. "New advanced k-w turbulence model for high-lift aerodynamics". *AIAA Paper 2004-1120*. Reno, Nevada. 2004.

[237] R. A. W. M. Henkes, F. F. van der Flugt, and C. J. Hoogendoorn. "Natural Convection Flow in a Square Cavity Calculated with Low-Reynolds-Number Turbulence Models". *Int. J. Heat Mass Transfer*. 34. 1543–1557. 1991.

[238] H. Hertz. “On the Evaporation of Liquids, Especially Mercury, in Vacuo”. *Annalen der Physik (Leipzig)*. 17. 177. 1882.

[239] H. Hertz. “Über die Berührung fester elastischer Körper”. *Journal für die reine und angewandte Mathematik*. 92. 156-171. 1881.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

[240] H. Hessenkemper, T. Ziegenhein, R. Rzehak, D. Lucas, and A. Tomiyama. “Lift force coefficient of ellipsoidal single bubbles in water”. *International Journal of Multiphase Flows*. 138. 2021.

[241] J. B. Heywood. *Internal Combustion Engine Fundamentals*. McGraw-Hill, New York, rev. edition. 1988.

[242] R. Herweg and R. R. Maly. A Fundamental Model for Flame Kernel Formation in S.I. Engines. SAE 922243. 1947-1976. 1992.

[243] T. Hibiki and M. Ishii. "One-group Interfacial Area Transport of Bubbly Flows in Vertical Round Tubes". *International Journal of Heat and Mass Transfer*. 43. 2711–2726. 2000.

[244] K. Higashitani, K. Yamauchi, Y. Matsuno, and G. Hosokawa. “Turbulent Coagulation of Particles Dispersed in a Viscous Fluid”. *Chemical Engineering Journal Japan*. 16(4). 299–304. 1983.

[245] P.G. Hill. “Condensation of Water Vapor During Supersonic Expansion in Nozzles”. *Journal of Fluid Mechanics*. 25 Part 3. 593–620. 1966.

[246] J. O. Hinze. *Turbulence*. McGraw-Hill Publishing Co., New York. 1975.

[247] J. O. Hirschfelder, C. F. Curtiss, and R. B. Bird. *Molecular Theory of Gases and Liquids*. John Wiley & Sons, New York. 1954.

[248] C. W. Hirt and B. D. Nichols. "Volume of Fluid (VOF) Method for the Dynamics of Free Boundaries". *J. Comput. Phys*. 39. 201–225. 1981.

[249] T. Höhne, Deendarlianto, D. Lucas. "Numerical simulations of counter-current two-phase flow experiments in a PWR hot leg model using an interfacial area density model". *Int. J. Heat Fluid Flow*. 32. 5. 1047–1056. 2011.

[250] T. Höhne, T. Geissler, A. Bieberle, U. Hampel. "Numerical modeling of a horizontal annular flow experiment using a droplet entrainment model". *Annals of Nuclear Energy*. 77. 351–360. 2015.

[251] T. Höhne, S. Haensch. "A droplet entrainment model for horizontal segregated flows". *Nuclear Eng. and Des.*. 286. 18–26. 2015.

[252] T. Höhne, P. Porombka. "Modelling horizontal two-phase flows using generalized models". *Annals of Nuclear Energy*. 111. 311–316. 2018.

[253] T. Höhne, P. Porombka, and S. Moya Saez. "Validation of AIAD sub-models for advanced numerical modelling of horizontal two-phase flows". *Fluids*. 5(3). 2020.

[254] D. G. Holmes and S. D. Connell. "Solution of the 2D Navier-Stokes Equations on Unstructured Adaptive Grids". *Presented at the AIAA 9th Computational Fluid Dynamics Conference*. June 1989.

[255] M. J. Hounslow, R. L. Ryall, and V. R. Marshall. “A Discretized Population Balance for Nucleation, Growth, and Aggregation”. *AIChE Journal*. 34(11). 1821–1832. 1988.

[256] T. J. Houser, M. Hull, R. Alway, and T. Biftu. *Int. Journal of Chem. Kinet*. 12. 579. 1980.

[257] S. Hosokawa, A. Tomiyama, S. Misaki, and T. Hamada. "Lateral Migration of Single Bubbles Due to the Presence of Wall". *ASME 2002 Joint U.S.-European Fluids Engineering Division Conference, Montreal, QC, Canada*. July 14–18, 2002.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

[258] M. A. Hsia and L. L. Tavlarides. “Simulation analysis of drop breakage, coalescence and micro-mixing in liquid–liquid stirred tanks”. *Chemical Engineering Journal*. 26(3). 189–199. 1983.

[259] P. Huang, P. Bradshaw, and T. Coakley. "Skin Friction and Velocity Profile Family for Compressible Turbulent Boundary Layers". *AIAA Journal*. 31(9). 1600–1604. September 1993.

[260] T. J. R. Hughes. *The Finite Element Method: Linear Static and Dynamic Finite Element Analysis*. Prentice-Hall, Inc. Englewood Cliffs, NJ. 1987.

[261] G. A. Hughmark. “Mass and heat transfer from rigid spheres”. *AICHE Journal*. 13. 1219–1221. November 1967.

[262] L. Huilin and D. Gidaspow. "Hydrodynamics of binary fluidization in a riser: CFD simulation using two granular temperatures”. *Chemical Engineering Science*. 58. 3777–3792. 2003.

[263] S. C. Hunter. "Formation of so3 in gas turbines". *Transactions of the ASME*. 104. 44–51. 1982.

[264] B. R. Hutchinson and G. D. Raithby. "A Multigrid Method Based on the Additive Correction Strategy". *Numerical Heat Transfer*. 9. 511–537. 1986.

[265] H. Ibdir and H. Arastoopour. "Modeling of multi-type particle flow using kinetic approach". *AICHE Journal*. May 2005.

[266] A. Ioilev, M. Samigulin and V. Ustinenko. "Advances in the Modeling of Cladding Heat Transfer and Critical Heat Flux in Boiling Water Reactor Fuel Assemblies". *The 12th International Topical Meeting on Nuclear Reactor Thermal Hydraulics (NURETH-12)*. Pittsburgh, Pennsylvania. 2007.

[267] M. Ihme, H. Pitsch. "Modeling of radiation and nitric oxide formation in turbulent nonpremixed flames using a flamelet/progress variable formulation". *Physics of Fluids*. 20. 5. 055110. AIP. 2008.

[268] K. Ishazaki, T. Ikohagi, and H. Daiguji. "A High-Resolution Numerical Method for Transonic Non-equilibrium Condensation Flows Through a Steam Turbine Cascade". *In Proceedings of the 6th International Symposium on Computational Fluid Dynamics*. 1. 479–484. 1995.

[269] M. Ishii. “Two-fluid model for two-phase flow”. *2nd International Workshop on Two-phase Flow Fundamentals*. RPI, Troy, NY. 1979.

[270] M. Ishii and M.A. Grolmes. "Inception Criteria for Droplet Entrainment in Two-Phase Concurrent Film Flow". *AIChE J.*. 21. 2. 308–318. 1975.

[271] M. Ishii and S. Kim. "Micro Four-Sensor Probe Measurement of Interfacial Area Transport for Bubbly Flow in Round Pipes". *Nuclear Engineering and Design*. 205. 123–131. 2001.

[272] M. Ishii and N. Zuber. "Drag Coefficient and Relative Velocity in Bubbly, Droplet or Particulate Flows". *AIChE J.*. 25. 843-855. 1979.

[273] R. I. Issa. "Solution of Implicitly Discretized Fluid Flow Equations by Operator Splitting". *Journal of Computational Physics*. 62. 40–65. 1986.

[274] S. Jain. "Three-Dimensional Simulation of Turbulent Particle Dispersion". *PhD thesis*. University of Utah, Utah. 1995.

[275] A. Jameson. "Solution of the Euler Equations for Two Dimensional Transonic Flow by a Multigrid Method". *MAE Report 1613*. Princeton University. June 1983.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

[276] A. Jameson, W. Schmidt, and E. Turkel. "Numerical Solution of the Euler Equations by Finite Volume Methods Using Runge-Kutta Time-Stepping Schemes". *Technical Report AIAA-81-1259*. AIAA 14th Fluid and Plasma Dynamics Conference, Palo Alto, California. June 1981.

[277] J. Janicka, W. Kolbe, and W. Kollmann. "Closure of the transport equation for the pdf of turbulent scalar fields". *Journal Non-Equilibrium Thermodynamics*. 4. 47. 1978.

[278] J. Janicka and W. Kollmann. "A Two-Variable Formulation for the Treatment of Chemical Reactions in Turbulent H_2-Air Diffusion Flames". *In 17th Symp. (Int'l.) on Combustion*. The Combustion Institute. 1978.

[279] J. Janicka and W. Kollmann. "A Numerical Study of Oscillating Flow Around a Circular Cylinder". *Combustion and Flame*. 44. 319–336. 1982.

[280] C. Jayatillaka. "The Influence of Prandtl Number and Surface Roughness on the Resistance of the Laminar Sublayer to Momentum and Heat Transfer". *Prog. Heat Mass Transfer*. 1. 193–321. 1969.

[281] P. C. Johnson and R. Jackson. "Frictional-Collisional Constitutive Relations for Granular Materials, with Application to Plane Shearing". *J. Fluid Mech*. 176. 67–93. 1987.

[282] W. P. Jones and J. H. Whitelaw. "Calculation Methods for Reacting Turbulent Flows: A Review". *Combustion and Flame*. 48. 1–26. 1982.

[283] T. Jongen. "Simulation and Modeling of Turbulent Incompressible Flows". *PhD thesis*. EPF Lausanne, Lausanne, Switzerland. 1992.

[284] T. Just and S. Kelm. *"Die Industry*. 38. 76. 1986.

[285] B. Kader. "Temperature and Concentration Profiles in Fully Turbulent Boundary Layers". *Int. J. Heat Mass Transfer*. 24(9). 1541–1544. 1981.

[286] B. Kakimpa, H. Morvan, S. Hibberd. "The depth-averaged numerical simulation of laminar thin-film flows with capillary waves". Open access Journal Publication. ASME. *J. Eng. Gas Turbines Power*. 138. 11. 112501–112501-10. DOI: 10.1115/1.4033471. 2016.

[287] B. Kakimpa, H. Morvan, S. Hibberd. "Solution Strategies for Thin Film Rimming Flow Modelling". Conference Paper. ASME. *Turbo Expo: Power for Land, Sea, and Air*. 5C: Heat Transfer. V05CT15A026. DOI: 10.1115/GT2015-43503. 2015.

[288] N. Kandamby, G. Lazopoulos, F. C. Lockwood, A. Perera, and L. Vigevano. "Mathematical Modeling of NOx Emission Reduction by the Use of Reburn Technology in Utility Boilers". *In ASME Int. Joint Power Generation Conference and Exhibition, Houston, Texas*. 1996.

[289] K. C. Karki and S. V. Patankar. "Pressure-Based Calculation Procedure for Viscous Flows at All Speeds in Arbitrary Configurations". *AIAA Journal*. 27. 1167–1174. 1989.

[290] S. Kase and T. Matsuo. “Studies on Melt Spinning. II. Steady-State and Transient Solution of Fundamental Equations Compared with Experimental Results”. *Journal of Applied Polymer Science*. 11. 251–287. 1967.

[291] I. Kataoka, M. Ishii, and K. Mishima. “Generation and size distribution of droplet in annular two-phase flow”. *J. of Fluids Engineering*. 105. 230–238. 1983.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

[292] M. Kato and B. E. Launder. “The modelling of turbulent flow around stationary and vibrating square cylinders”. *Ninth Symposium on "Turbulent Shear Flows"*. Kyoto, Japan: August 16-18, 1993.

[293] A. Kaufmann, M. Moreau, O. Simonin, J. Helie. “Comparison between Lagrangian and mesoscopic Eulerian modelling approaches for inertial particles suspended in decaying isotropic turbulence”. *Journal of Computational Physics*. 277. 13. 6448–6472. Academic Press Professional, Inc., San Diego, CA, USA. June, 2008.

[294] W. M. Kays. "Loss coefficients for abrupt changes in flow cross section with low reynolds number flow in single and multiple tube systems". *Transactions of the ASME*. 72. 1067–1074. January 1950.

[295] W. M. Kays. "Turbulent Prandtl Number - Where Are We?". *J. Heat Transfer*. 116. 284–295. 1994.

[296] W. M. Kays and A. L. London. *Compact Heat Exchangers*. McGraw-Hill, New York. 1964.

[297] A. Kazakov, M. Frenklach. "Dynamic Modeling of Soot Particle Coagulation and Aggregation: Implementation With the Method of Moments and Application to High-Pressure Laminar Premixed Flames". *Combustion and Flame*. 114. 484–501. 1998.

[298] R. J. Kee, M.E. Coltrin, and P. Glarborg. *Chemically Reacting Flow: Theory and Practice*. Wiley. 2003.

[299] R. J. Kee, F. M. Rupley, J. A. Miller, M. E. Coltrin, J. F. Grcar, E. Meeks, H. K. Moffat, A. E. Lutz, G. Dixon-Lewis, M. D. Smooke, J. Warnatz, G. H. Evans, R. S. Larson, R. E. Mitchell, L. R. Petzold, W. C. Reynolds, M. Caracotsios, W. E. Stewart, P. Glarborg, C. Wang, O. Adigun, W. G. Houf, C. P. Chou, S. F. Miller, P. Ho, and D. J. Young. CHEMKIN v. 4.0*Technical Report*. Reaction Design, Inc., San Diego, CA. 2004.

[300] I. M. Khan and G. Greeves. "A Method for Calculating the Formation and Combustion of Soot in Diesel Engines". In N. H.Afgan and J. M.Beer, editors*Heat Transfer in Flames*. Chapter 25Scripta, Washington DC. 1974.

[301] U. S. Kim, C. B. Shin, and C.-S. Kim. “Effect of Electrode Configuration on the Thermal Behavior of a Lithium-Polymer Battery”. *Journal of Power Sources*. Princeton, NJ. 180. 909–916. 2008.

[302] J. S. Kim and F. A. Williams. "Extinction of Diffusion Flames with Non-Unity Lewis Number". *Eng. Math*. 31. 101–118. 1997.

[303] S. Kim, D. Caraeni, and B. Makarov. "A Multidimensional Linear Reconstruction Scheme for Arbitrary Unstructured Grids". *Technical report*. AIAA 16th Computational Fluid Dynamics Conference, Orlando,FloridaAmerican Institute of Aeronautics and Astronautics. June 2003.

[304] S. Kim, Q. Wu, and M. Ishii. "One-group Interfacial Area Transport in Vertical Bubbly Flow". *International Journal of Heat and Mass Transfer*. 41. 1103–1112. 1997.

[305] S.-E. Kim. "Large eddy simulation using unstructured meshes and dynamic subgrid-scale turbulence models". *Technical Report AIAA-2004-2548*. 34th Fluid Dynamics Conference and ExhibitAmerican Institute of Aeronautics and Astronautics,. June 2004.

[306] S.-E. Kim and D. Choudhury. "A Near-Wall Treatment Using Wall Functions Sensitized to Pressure Gradient". In ASME FED Vol. 217,*Separated and Complex Flows*. ASME. 1995.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

[307] S.-E. Kim, D. Choudhury, and B. Patel. "Computations of Complex Turbulent Flows Using the Commercial Code Ansys Fluent". *In Proceedings of the ICASE/LaRC/AFOSR Symposium on Modeling Complex Turbulent Flows*. Hampton, Virginia1997.

[308] W.-W. Kim and S. Menon. "Application of the localized dynamic subgrid-scale model to turbulent wall-bounded flows". *Technical Report AIAA-97-0210*. 35th Aerospace Sciences Meeting, Reno, NVAmerican Institute of Aeronautics and Astronautics. January 1997.

[309] G-H. Kim, A. Pesaran, R. Spotnitz. "A Three-dimensional thermal abuse model for lithium-ion cells". *Journal of Power Sources*. 170 (2). 476-489. 2007.

[310] G-H. Kim et al. "Multi-Domain Modeling of Lithium-Ion Batteries Encompassing Multi-Physics in Varied Length Scales". *J. of Electrochemical Soc.*. 158 (8). A955-A969. 2011.

[311] U. S. Kim et al. "Modeling the Dependence of the Discharge Behavior of a Lithium-Ion Battery on the Environmental Temperature". *J. of Electrochemical Soc.*. 158 (5). A611-A618. 2011.

[312] U. S. Kim et al. "Effect of electrode configuration on the thermal behavior of a lithium-polymer battery". *Journal of Power Sources*. 180 (2). 909-916. 2008.

[313] M. Knudsen. "Maximum Rate of Vaporization of Mercury". *Annalen der Physik (Leipzig)*. 47. 697. 1915.

[314] H. Kobayashi, J. B. Howard, and A. F. Sarofim. "Coal Devolatilization at High Temperatures". *In 16th Symp. (Int'l.) on Combustion*. The Combustion Institute. 1976.

[315] G. Kocamustafaogullari and M. Ishii. "Interfacial area and nucleation site density in boiling systems". *International Journal of Heat and Mass Transfer*. 26. 9. 1377–1387. 1983.

[316] G. Kocamustafaogullari and M. Ishii. "Foundation of the Interfacial Area Transport Equation and its Closure Relations". *International Journal of Heat and Mass Transfer*. 38. 481–493. 1995.

[317] N. I. Kolev. *Multiphase Flow Dynamics 2: Thermal and Mechanical Interactions*. Springer, Berlin, Germany, 2nd edition. 2005.

[318] S. C. Kong, Z. W. Han, and R. D. Reitz. "The Development and Application of a Diesel Ignition and Combustion Model for Multidimensional Engine Simulations". *SAE Paper 950278*. SAE. 1995.

[319] M. Kostoglou, S. Dovas, and A. J. Karabelas. "On the Steady-State Size Distribution of Dispersions in Breakage Processes". *Chemical Engineering Science*. 52(8). 1285–1299. 1997.

[320] R. Kraichnan. "Diffusion by a Random Velocity Field". *Physics of Fluids*. 11. 21–31. 1970.

[321] J. C. Kramlich. "The Fate and Behavior of Fuel-Sulfur in Combustion Systems". *PhD thesis*. Washington State University, Washington, USA. 1980.

[322] D. Kuhnke. "Spray Wall Interaction Modelling by Dimensionless Data Analysis". *PhD thesis*. Fachbereich Mathematik, Technischen Universität Darmstadt. 2004.

[323] A. A. Kulikovsky, J. Divisek, and A. A. Kornyshev. "Modeling the Cathode Compartment of Polymer Electrolyte Fuel Cells: Dead and Active Reaction Zones". *Journal, Electrochemical Society*. 146(11). 3981–3991. 1999.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

[324] S. Kumar and D. Ramkrishna. "On the Solution of Population Balance Equations by Discretization - I., A Fixed Pivot Technique". *Chemical Engineering Science*. 51(8). 1311–1332. 1996.

[325] K. K. Y. Kuo. *Principles of Combustion*. John Wiley and Sons, New York. 1986.

[326] N. Kurul and M. Z. Podowski. "On the modeling of multidimensional effects in boiling channels". *In Proceedings of the 27th National Heat Transfer Conference, Minneapolis, Minnesota, USA*. 1991.

[327] S. S. Kutateladze. "Boiling Heat Transfer". *International Journal of Heat and Mass Transfer*. 4. 31–45. 1961.

[328] K. H. Kwon et al. "A Two-dimensional Modeling of a Lithium-polymer battery". *Journal of Power Sources*. 163. 151-157. 2006.

[329] H. Lamb. *Hydrodynamics, Sixth Edition*. Dover Publications, New York. 1945.

[330] M.J. Lampinen, M. Fomino. "Analysis of free energy and entropy changes for half-cell reactions". *J. Electrochem. Soc.*. 140(12). 3537-3546. 1993.

[331] G. L. Lane, M. P. Schwarz, and G. M. Evans. Modelling of the Interaction between Gas and Liquid in Stirred Vessels. *10th European Conference on Mixing*. 197–204. 2000.

[332] R. B. Langtry, F. R. Menter, S. R. Likki, Y. B. Suzen, P. G. Huang, and S. Volker. "A Correlation-Based Transition Model Using Local Variables: Part II — Test Cases and Industrial Applications". (ASME-GT2004-53454) 2004.

[333] R. B. Langtry and F. R. Menter. "Correlation-Based Transition Modeling for Unstructured Parallelized Computational Fluid Dynamics Codes". *AIAA Journal*. 47(12). 2894–2906. December 2009.

[334] M. E. Larsen and J. R. Howell. "Least Squares Smoothing of Direct Exchange Areas in Zonal Analysis". *J. Heat Transfer*. 108. 239–242. 1986.

[335] B. E. Launder. "Second-Moment Closure and Its Use in Modeling Turbulent Industrial Flows". *International Journal for Numerical Methods in Fluids*. 9. 963–985. 1989.

[336] B. E. Launder. "Second-Moment Closure: Present... and Future?". *Inter. J. Heat Fluid Flow*. 10(4). 282–300. 1989.

[337] B. E. Launder, G. J. Reece, and W. Rodi. "Progress in the Development of a Reynolds-Stress Turbulence Closure". *J. Fluid Mech*. 68(3). 537–566. April 1975.

[338] Launder B. E. and Sharma B. I.. (1974): "Application of the Energy-Dissipation Model of Turbulence to the Calculation of Flow near a Spinning Disc". Letters in Heat and Mass Transfer, 1:131–138..

[339] B. E. Launder and N. Shima. "Second-Moment Closure for the Near-Wall Sublayer: Development and Application". *AIAA Journal*. 27(10). 1319–1325. 1989.

[340] B. E. Launder and D. B. Spalding. *Lectures in Mathematical Models of Turbulence*. Academic Press, London, England. 1972.

[341] B. E. Launder and D. B. Spalding. "The Numerical Computation of Turbulent Flows". *Computer Methods in Applied Mechanics and Engineering*. 3. 269–289. 1974.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

[342] N. M. Laurendeau. "Heterogeneous Kinetics of Coal Char Gasification and Combustion". *Prog. Energy Comb. Sci*. 4. 221–270. 1978.

[343] J. Lavieville, E. Quemerais, S. Mimouni, M. Boucker and N. Mechitoua. "NEPTUNE CFD V1.0 Theory Manual". *EDF*. 2005.

[344] C. K. Law and H. K. Law . "A d2 – Law for Multicomponent Droplet Vaporization and Combustion". *AIAA*. Vol. 20. No. 4. 552–527. 1982.

[345] J. L. Lebowitz. "Exact Solution of Generalized Percus-Yevick Equation for a Mixture of Hard Spheres". *The Phy. Rev*. 133(4A). A895–A899. 1964.

[346] J. Lee, A .Alam, and H. Ju. "Multidimensional and transient modeling of an alkaline water electrolysis cell". *International Journal of Hydrogen Energy* . 46(26). 13678–13690. 2021.

[347] K. B. Lee, M. W. Thring, and J. M. Beer. "On the Rate of Combustion of Soot in a Laminar Soot Flame". *Combustion and Flame*. 6. 137–145. 1962.

[348] W. H. Lee. "A Pressure Iteration Scheme for Two-Phase Modeling". *Technical Report LA-UR 79-975*. Los Alamos Scientific Laboratory, Los Alamos, New Mexico. 1979.

[349] A. H. Lefebvre. *Atomization and Sprays*. Hemisphere Publishing Corporation. 1989.

[350] D. Legendre and J. Magnaudet. “The Lift Force on a Spherical Bubble in a Viscous Linear Shear Flow”. *J. Fluid Mech*. 368. 81–126. 1998.

[351] F. Lehr, M. Millies, and D. Mewes. “Bubble-Size Distributions and Flow Fields in Bubble Columns”. *AIChE Journal*. 8(11). 2426–2443. 2002.

[352] M. Lemmert and L. M. Chawla. *Influence of flow velocity on surface boiling heat transfer coefficient in Heat Transfer in Boiling*. E. Hahne and U. Grigull, Eds., Academic Press and Hemisphere, New York, NY, USA. 1977.

[353] B. P. Leonard. "The ULTIMATE conservative difference scheme applied to unsteady one-dimensional advection". *Comp. Methods Appl. Mech. Eng*. 88. 17–74. 1991.

[354] B. P. Leonard and S. Mokhtari. "ULTRA-SHARP Nonoscillatory Convection Schemes for High-Speed Steady Multidimensional Flow". *NASATM1-2568 (ICOMP-90-12)*. NASA Lewis Research Center. 1990.

[355] K. M. Leung and R. P. Lindsted. "Detailed Kinetic Modeling of C1-C3 Alkane Diffusion Flames". *Combustion and Flame*. 102. 129–160. 1995.

[356] M. Leverett. “Capillary behaviour in porous solids”. *Transactions of the AIME*. 142. 152–169. 1941.

[357] V. G. Levich. *Physicochemical Hydrodynamics*. Prentice Hall. 1962.

[358] J. M. Levy, L. K. Chen, A. F. Sarofim, and J. M. Beer. "NO/Char Reactions at Pulverized Coal Flame Conditions". *In 18th Symp. (Int'l.) on Combustion*. The Combustion Institute. 1981.

[359] M.T. Lewandowski and I.S. Ertesvag. "Analysis of the Eddy Dissipation Concept formulation for MILD combustion modelling". *Fuel*. 224. 687–700. 2018.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

[360] M.T. Lewandowski, Z. Li, A. Parente, and J. Pozorski . "Generalised Eddy Dissipation Concept for MILD combustion regime at low local Reynolds and Damköhler numbers. Part 2: Validation of the model". *Fuel*. 278. 117743. 2020.

[361] A. Li and G. Ahmadi. "Dispersion and Deposition of Spherical Particles from Point Sources in a Turbulent Channel Flow". *Aerosol Science and Technology*. 16. 209–226. 1992.

[362] D. Li , S. Venkateswaran, K. Fakhari and C. Merkle. "Convergence Assessment of General Fluid Equations on Unstructured Hybrid Grids". *Technical Report AIAA-2001-2557*. AIAA 15th Computational Fluid Dynamics Conference, Anaheim, CaliforniaAmerican Institute of Aeronautics and Astronautics. June 2001.

[363] X. Li and R. S. Tankin. "On the Temporal Instability of a Two-Dimensional Viscous Liquid Sheet". *Journal of Fluid Mechanics*. 226. 425. 1991.

[364] L. Liang, J. G. Stevens, J. T Farrell. "A Dynamic Adaptive Chemistry Scheme for Reactive Flow Computations". *Proceedings of the Combustion Institute*. 32. 527-534. 2009.

[365] L. Liang, J. G. Stevens, et al.. "The Use of Dynamic Adaptive Chemistry in Combustion Simulation of Gasoline Surrogate Fuels". *Combustion and Flame*. 156. 7. 1493-1502. 2009.

[366] Y. Liao, R. Rzehak, D. Lucas, E. Krepper. "Baseline closure model for dispersed bubbly flow: Bubble coalescence and breakup". *Chem. Eng. Sci.*. 122. 27. 336-349. 2015.

[367] A. K. Lichtarowicz, R. K. Duggins, and E. Markland. "Discharge Coefficients for Incompressible Non-Cavitating Flow Through Long Orifices". *Journal of Mechanical Engineering Science*. 7. 2. 1965.

[368] F. S. Lien and M. A. Leschziner. "Assessment of Turbulent Transport Models Including Non-Linear RNG Eddy-Viscosity Formulation and Second-Moment Closure". *Computers and Fluids*. 23(8). 983–1004. 1994.

[369] J. H. Lienhard IV and J. H. Lienhard V. *A Heat Transfer Textbook*. Version 5.10. Phlogiston Press, Cambridge. 784 pages. 2020.

[370] J.H. Lienhard V. "Heat Transfer by Impingement of Circular Free Surface Liquid Jets". *18th National & 7th ISHMT-ASME Heat and Mass Transfer Conference*. 2006.

[371] M. J. Lighthill. "On Sound Generated Aerodynamically". *Proc. Roy. Soc. London*. A211564–587. 1952.

[372] G. M. Lilley. "The radiated noise from isotropic turbulence revisited". *NASA Contract Report 93-75*. NASA Langley Research Center, Hampton, VA. 1993.

[373] D. K. Lilly. "A Proposed Modification of the Germano Subgrid-Scale Closure Model". *Physics of Fluids*. 4. 633–635. 1992.

[374] F. Lindemann. *Trans. Faraday Soc*. 7. 1922.

[375] R. P. Lindstedt. *In Proc. IUTAM Conf. on Aerothermo-Chemistry in Combustion*. Taipei, Taiwan1991.

[376] R. P. Lindstedt. *"Soot Formation in Combustion*. Springer-Verlag, Berlin. 1994.

[377] M. S. Liou. "A sequel to AUSM: AUSM+". *Journal of Computational Physics*. 129. 364–382. 1996.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

[378] M. S. Liou and C. J. Steffen, Jr. "A new flux splitting scheme". *Journal of Computational Physics*. 107(1). 23–39. 1993.

[379] A. N. Lipatnikov and J. Chomiak. "Turbulent Flame Speed and Thickness: Phenomenology, Evaluation and Application in Multi-dimensional Simulations". *Progress in Energy& Combustion Science*. 28. 1–74. January 2002.

[380] R. J. Litchford and S. M. Jeng. "Efficient Statistical Transport Model for Turbulent Particle Dispersion in Sprays". *AIAA Journal*. 29. 1443. 1991.

[381] J. D. Litster, D. J. Smit, and M. J. Hounslow. "Adjustable Discretization Population Balance for Growth and Aggregation". *AIChE Journal*. 41(3). 591–603. 1995.

[382] A. B. Liu, D. Mather, and R. D. Reitz. "Modeling the Effects of Drop Drag and Breakup on Fuel Sprays". *SAE Technical Paper 930072*. SAE. 1993.

[383] H. Liu and B. M. Gibbs. "Modeling of NO and N_2O Emissions from Biomass-Fired Circulating Fluidized Bed Combustors". *Fuel*. 81. 271–280. 2002.

[384] F. C. Lockwood and C. A Romo-Millanes. "Mathematical Modelling of Fuel - NO Emissions From PF Burners". *J. Int. Energy*. 65. 144–152. September 1992.

[385] G. Lodier, L. Vervisch, V. Moureau and P. Domingo. "Composition-Space Premixed Flamelet Solution with Differential Diffusion for In Situ Flamelet-Generated Manifolds". *Combustion and Flame*. 158. 10. 2009-2016. 2011.

[386] M. Lopez de Bertodano. "Turbulent Bubbly Flow in a Triangular Duct". *Ph.D. Thesis*. Rensselaer Polytechnic Institute, Troy, New York. 1991.

[387] M. A. Lopez de Bertodano, C. S. Jan, S. G. Beus. "Annular Flow Entrainment Rate Experiment in a Small Vertical Pipe". *Nuclear Engineering and Design*. 178. 61-70. 1997.

[388] B. Lu, W. Wang, and J. Li. "Searching for a mesh-independent sub-grid model for CFD simulation of gas–solid riser flows". *Chemical Engineering Science*. 64. 15. 3437-3447 . 2009.

[389] T. F. Lu and C. K. Law. "A Directed Relation Graph Method for Mechanism Reduction". Proc. Combust. Inst. 30. 1. 1333-1341. 2005.

[390] T. F. Lu and C. K. Law. "On the Applicability of Directed Relation Graph to the Reduction of Reaction Mechanisms". *Combust. Flame*. 146(3). 472-483. 2006.

[391] T. F. Lu and C. K. Law. "Toward Accommodating Realistic Fuel Chemistry in Large-Scale Computation". *Prog. Energy Combust. Sci*. 35(1). 192-215. 2009.

[392] N. Lubchenko. "Near-Wall Modeling of Bubbly Flows". *PhD thesis*. Massachusetts Institute of Technology. 2015.

[393] D. Lucas, T. Frank, C. Lifante, P. Zwart, and A. Burns . "Extension of the inhomogeneous MUSIG model for bubble condensation". *Nuclear Engineering and Design*. 241. 11. 4359–4367. 2011.

[394] C. K. K. Lun, S. B. Savage, D. J. Jeffrey, and N. Chepurniy. "Kinetic Theories for Granular Flow: Inelastic Particles in Couette Flow and Slightly Inelastic Particles in a General Flow Field". *J. Fluid Mech*. 140. 223–256. 1984.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

[395] H. Luo. “Coalescence, Breakup and Liquid Circulation in Bubble Column Reactors”. *PhD thesis from the Norwegian Institute of Technology*. Trondheim,Norway1993.

[396] J. Y. Luo, R. I. Issa, and A. D. Gosman. "Prediction of Impeller-Induced Flows in Mixing Vessels Using Multiple Frames of Reference". *In IChemE Symposium Series*. 136. 549–556. 1994.

[397] H. Luo and H. F. Svendsen. “Theoretical Model for Drop and Bubble Breakup in Turbulent Dispersions”. *AIChE Journal*. 42(5). 1225–1233,. 1996.

[398] A. E. Lutz, R. J. Kee, J. F. Grcar, and F. M. Rupley. "OPPDIF: A FORTRAN Program for Computing Opposed-Flow Diffusion Flames". *Sandia National Laboratories Report SAND96-8243*. 1997.

[399] J. F. Lynn. "Multigrid Solution of the Euler Equations with Local Preconditioning". *PhD thesis*. University of Michigan. 1995.

[400] R. K. Lyon. "The NH_3-NO-O O_2 Reaction". *Int. Journal of Chem. Kinetics*. 8. 315–318. 1976.

[401] D. Ma and G. Ahmadi. "A Thermodynamical Formulation for Dispersed Multiphase Turbulent Flows". *Int.J.Multiphase Flow*. 16. 323–351. 1990.

[402] J. Ma. "Soot Formation and Soot Secondary Reactions During Coal Pyrolysis". *PhD thesis*. Brigham Young University, Utah, USA. 1996.

[403] D. D. MacNeil and J. R. Dahn. "Test of Reaction Kinetics Using Both Differential Scanning and Accelerating Rate Calorimetries As Applied to the Reaction of LixCoO2 in Non-aqueous Electrolyte". *J. Phys. Chem*. A 105. 4430–4439. 2001.

[404] R. K. Madabhushi. "A Model for Numerical Simulation of Breakup of a Liquid Jet in Crossflow". *"Atomization and Sprays"*. 13. 413–424. 2003.

[405] J. I. Madsen, W. A. Rogers, and T. J. OBrien. "Unstructured Multigrid Through Agglomeration In Computational Modelling of Mercury Control by Sorbent Injection". Baltimore, Maryland 2004.

[406] B. F. Magnussen. "On the Structure of Turbulence and a Generalized Eddy Dissipation Concept for Chemical Reaction in Turbulent Flow". Nineteenth AIAA Meeting, St. Louis 1981.

[407] B. F. Magnussen and B. H. Hjertager. "On mathematical models of turbulent combustion with special emphasis on soot formation and combustion". *In 16th Symp. (Int'l.) on Combustion*. The Combustion Institute. 1976.

[408] M. Manninen, V. Taivassalo, and S. Kallio. "On the mixture model for multiphase flow". VTT Publications 288, Technical Research Centre of Finland. 1996.

[409] D. L. Marchisio and R. O. Fox. "Solution of Population Balance Equations Using the Direct Quadrature Method of Moments". *Aerosol Science and Technology*. 36. 43–73. 2005.

[410] D. L. Marchisio, R. D. Virgil, and R. O. Fox. “Quadrature Method of Moments for Aggregation-Breakage Processes”. *Journal of Colloid and Interface Science*. 258. 322–334. 2003.

[411] F. Mathey, D. Cokljat, J. P. Bertoglio, and E. Sergent. "Specification of LES Inlet Boundary Condition Using Vortex Method". In K. Hanjali , Y. Nagano, and M. Tummers, editors*4th International Symposium on Turbulence, Heat and Mass Transfer, Antalya, Turkey*. Begell House, Inc. 2003.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

[412] S. R. Mathur and J. Y. Murthy. "Coupled ordinates method for multigrid acceleration of radiation calculations". *J. of Thermophysics and Heat Transfer*. 13(4). 1783-1785. 1999.

[413] C. Martínez-Bazán, J. L. Montañés, J. C. Lasheras. "On the breakup of an air bubble injected into a fully developed turbulent flow. Part 1. Breakup frequency". *J. Fluid Mech.*. 401. 157-182. 1999.

[414] S. Mayer. "Theory of Liquid Atomization in High Velocity Gas Streams". *ARS Journal*. 31. 467–473. 1961.

[415] S. Majumdar. "Role of Underrelaxation in Momentum Interpolation for Calculation of Flow With Nonstaggered Grids". *Numerical Heat Transfer*. 13(1). 125–132. 1988.

[416] S. Mazumder and J. V. Cole . "Rigorous 3-D Mathematical Modeling of PEM Fuel Cells II. Model Predictions with Liquid Water Transport". *Journal, Electrochemical Society*. 150(11). A1510–1517. 2003.

[417] B. J. McCoy and M. Wang. "Continuous-mixture Fragmentation Kinetics Particle Size Reduction and Molecular Cracking". *Chemical Engineering Science*. 49(22). 3773–3785. 1994.

[418] H. A. McGee. *Molecular Engineering*. McGraw-Hill, New York. 1991.

[419] R. McGraw. "Description of Aerosol Dynamics by the Quadrature Method of Moments". *Aerosol Science and Technology*. 27. 255–265. 1997.

[420] B. S. McLaury et al. "Modeling erosion in chokes". *Proceeding of ASME Fluids Eng. Summer Meeting*. San Diego, California. 1996.

[421] B. S. McLaury, J. Wang, S. A. Shirazi, J. R. Shadley, and E. F. Rybicki. "Solid Particle Erosion in Long Radius Elbows and Straight Pipes". *SPE Annual Technical Conference and Exhibition, II Production Operations and Engineering/General, San Antonio, Texas*. SPE Paper 38842 October 1997.

[422] R. Mei and J. F. Klausner. "Shear Lift Force on Spherical Bubbles". *J.Fluid Mech*. 15. 62–65. 1994.

[423] P. C. Melte and D. T. Pratt. "Measurement of Atomic Oxygen and Nitrogen Oxides in Jet Stirred Combustion". *In 15th Symposium (Int'l) on Combustion*. The Combustion Institute. 1061–1070. 1974.

[424] C. Meneveau and T. Poinsot. "Stretching and Quenching of Flamelets in Premixed Turbulent Combustion". *Combustion and Flame*. 86. 311–332. 1991.

[425] F. R. Menter. "Two-Equation Eddy-Viscosity Turbulence Models for Engineering Applications". *AIAA Journal*. 32(8). 1598–1605. August 1994.

[426] F. R. Menter. "Review of the SST Turbulence Model Experience from an Industrial Perspective". *International Journal of Computational Fluid Dynamics*. Volume 23, Issue 4. 2009.

[427] F. R. Menter and Y. Egorov. "Re-visiting the Turbulent Scale Equation". *Proc. IUTAM Symp. One Hundred Years of Boundary Layer Research, Göttingen*. Springer. 2004.

[428] F. Menter and Y. Egorov. The Scale-Adaptive Simulation Method for Unsteady Turbulent Flow Predictions. Part 1: Theory and Model Description. *Journal Flow Turbulence and Combustion*. 85. 113–138. 2010.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

[429] F. R. Menter, M. Kuntz, and R. Langtry. "Ten Years of Experience with the SST Turbulence Model". In K. Hanjalic, Y. Nagano, and M. Tummers, editors*Turbulence, Heat and Mass Transfer*. 4. Begell House Inc. 625–632. 2003.

[430] F. R. Menter, R. B. Langtry, S. R. Likki, Y. B. Suzen, P. G. Huang, and S. Volker. "A Correlation-Based Transition Model Using Local Variables: Part I — Model Formulation". (ASME-GT2004-53452) 2004.

[431] F. R. Menter, . M. Kuntz and R. Bender. "A scale-adaptive simulation model for turbulent flow predictions". *AIAA Journal*. January 2003.

[432] K. V. Meredith, A. Heather, J. de Vries and Y. Xin. "A Numerical Model for Partially-Wetted Flow of Thin Liquid Films". *Computational Methods in Multiphase Flow VI*. 70. 239–250. WIT Press. 2011.

[433] H. J. Merk. "The Macroscopic Equations for Simultaneous Heat and Mass Transfer in Isotropic, Continuous and Closed Systems". *Appl. Sci. Res*. 8. 73–99. 1958.

[434] M. Metghalchi and J. C. Keck. "Burning velocities of mixtures of air with methanol, isooctane and indolene at high pressures and temperatures". *Combustion and Flame*. 48. 191–210. 1982.

[435] C.C. Milioli, F.E. Milioli, W. Holloway, K. Agrawal, S. Sundaresan. "Filtered two-fluid models of fluidized gas particle flows: new constitutive relations". *AIChE Journal*. 59. 9. 3265–3275. 2013.

[436] J. A. Miller and C. T. Bowman. "Mechanism and Modeling of Nitrogen Chemistry in Combustion". *Prog. in Energy and Comb. Sci*. 15. 287–338. 1989.

[437] J. A. Miller, M. C. Branch, W. J. McLean, D. W. Chandler, M. D. Smooke, and R. J. Kee. *In 20th Symp. (Int'l.) on Combustion*. The Combustion Institute. 673. 1985.

[438] J. A. Miller and G. A. Fisk. *Chemical and Engineering News*. 31. 1987.

[439] R. S. Miller, K. Harstad and J. Bellan. "Evaluation of Equilibrium and Non-Equilibrium Evaporation Models for Many Droplet Gas-Liquid Flow Simulations". *International Journal of Multiphase Flow*. 24. 6. 1025 –1055. 1998.

[440] M. Millies, D. Mewes. "Interfacial area density in bubbly flow". *Chemical Engineering and Processing*. 38. 4-6. 307–319. 1999.

[441] M. Missaghi. "Mathematical Modelling of Chemical Sources in Turbulent Combustion". *PhD thesis*. The University of Leeds, England. 1987.

[442] M. Missaghi, M. Pourkashanian, A. Williams, and L. Yap. *In Proceedings of American Flame Days Conference, USA*. 1990.

[443] M. F. Modest. "The Weighted-Sum-of-Gray-Gases Model for Arbitrary Solution Methods in Radiative Transfer". *J. Heat Transfer*. 113. 650–656. 1991.

[444] M. F. Modest. *Radiative Heat Transfer*. Series in Mechanical EngineeringMcGraw-Hill. 1993.

[445] M. F. Modest. *Radiative Heat Transfer 2nd Edition*. Academic Press. New York, . 2003.

[446] L. J. Molero de Blas. "Pollutant Formation and Interaction in the Combustion of Heavy Liquid Fuels". *PhD thesis*. University of London, England. 1998.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

[447] J. P. Monat, R. K. Hanson, and C. H. Kruger. *In 17th Symp. (Int'l.) on Combustion*. The Combustion Institute. 543. 1979.

[448] G. Montante, G. Micale, F. Magelli, and A. Brucato. *Experiments and CFD predictions of solid particle distribution in a vessel agitated with four pitched blade turbines*. Chemical Engineering Research and Design. 79(8). 1005-1010. 2001.

[449] G. Montoya. "Development and Validation of Advanced Theoretical Modeling for Churn-Turbulent Flows and Subsequent Transitions". *PhD Thesis*. Technischen Universität Berlin. 2015.

[450] G. Montoya, E. Baglietto, D. Lucas. "Implementation and Validation of a Surface Tension Model for the Multi-Scale Approach GENTOP". *16th International Topical Meeting on nuclear Reactor Thermalhydraulics - NURETH-16*. Chicago, USA. September 2015.

[451] G. Montoya, E. Baglietto, D. Lucas, E. Krepper, T. Höhne. "Comparative Analysis of High Void Fraction Regimes Using an Averaging Euler-Euler Multi-field Approach and a Generalized Two-Phase Flow (GENTOP) Concept". *22sd International Conference on Nuclear Engineering (ICONE22)*. Prague, Czech Republic. July 2014.

[452] G. Montoya, Deendarlianto, D. Lucas, T. Höhne, C. Vallée. "Image-Processing Based Study of the Interfacial Behavior of the Countercurrent Gas-Liquid Two-Phase Flow in a Hot Leg of a PWR". *Sci. Tech. Nuclear Int.*. Article ID: 209542 . 2012.

[453] G. Montoya, D. Lucas, E. Krepper, S. Hänsch, E. Baglietto. "Analysis and Applications of a Generalized Multi-Field Two-Fluid Approach for Treatment of Multi-Scale Interfacial Structures in High Void Fraction Regimes". *2014 International Congress on Advances in Nuclear Power Plants (ICAPP 2014)*. Charlotte, North Carolina, USA. April 2014.

[454] M. J. Moore and C. H. Sieverding. *"Two-Phase Steam Flow in Turbines and Separator*. McGraw-Hill. 1976.

[455] F. J. Moraga, R. T. Bonetto and R. T. Lahey. "Lateral forces on spheres in turbulent uniform shear flow". *International Journal of Multiphase Flow*. 25. 1321–1372. 1999.

[456] R. Moreau. *Magnetohydrodynamics*. Kluwer Academic Publishers. 1990.

[457] R. Moreno-Atanasio and M. Ghadiri. "Mechanistic Analysis and Computer Simulation of Impact Breakage of Agglomerates Effect of Surface Energy". *Chemical Engineering Science*. 61(8). 2476–2481. 2006.

[458] S. A. Morsi and A. J. Alexander. "An Investigation of Particle Trajectories in Two-Phase Flow Systems". *J. Fluid Mech*. 55(2). 193–208. September 26 1972.

[459] S. A. Morsi and A. J. Alexander. "An Investigation of Particle Trajectories in Two-Phase Flow Systems". *J. Fluid Mech*. 55(2). 193–208. September 26 1972.

[460] J. B. Moss, S. A. L. Perera,C. D. Stewart, and M. Makida. "Radiation Heat Transfer in Gas Turbine Combustors". *In Proc. 16th (Int'l.) Symp. on Airbreathing Engines, Cleveland, OH*. 2003.

[461] C. M. Muller, H. Breitbach, and N. Peters. "Partially Premixed Turbulent Flame Propagation in Jet Flames". *In 25th Symposium (Int) on Combustion*. The Combustion Institute. 1994.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

[462] C. Mundo, M. Sommerfeld, and C. Tropea. "Droplet-Wall Collisions: Experimental Studies of the Deformation and Breakup Process". *International Journal of Multiphase Flow*. 21(2). 151–173. 1995.

[463] J. Y. Murthy and S. R. Mathur. "A Finite Volume Method For Radiative Heat Transfer Using Unstructured Meshes". *AIAA-98-0860*. January 1998.

[464] S. Muzaferija, M. Peric, P. Sames, and T. Schellin. "A Two-Fluid Navier-Stokes Solver to Simulate Water Entry". *In Proc 22nd Symposium on Naval Hydrodynamics, Washington, DC*. 277–289. 1998.

[465] J. D. Naber and R. D. Reitz. "Modeling Engine Spray/Wall Impingement". *Technical Report 880107, Society of Automotive Engineers*. General Motors Research Laboratories, Warren, MI. 1988.

[466] J. H. Nam and M. Karviany. "Effective Diffusivity and Water-Saturation Distribution in Single- and Two-layer PEMFC Diffusion Medium". *Int. Journal Heat Mass Transfer*. 2003.

[467] M. Namazian and J. B. Heywood. "Flow in the Piston Cylinder Ring Crevices of a Spark Ignition Engine: Effect on Hydrocarbon Emissions, Efficiency, and Power". *SAE Technical Paper 820088*. SAE. 1982.

[468] I. Naruse, Y. Yamamoto, Y. Itoh, and K. Ohtake. "Fundamental Study on N_2O Formation/Decomposition Characteristics by Means of Low-Temperature Pulverized Coal Combustion". *In 26th Symposium (Int'l) on Combustion*. The Combustion Institute. 3213–3221. 1996.

[469] P. F. Nelson, A. N. Buckley, and M. D. Kelly. "Functional Forms of Nitrogen in Coals and the Release of Coal Nitrogen as NOx Precursors (HCN and NH_3)". *In 24th Symposium (Int'l) on Combustion*. The Combustion Institute. 1259. 1992.

[470] J. S. Newman. *Electrochemical Systems*. Prentice Hall. Englewood Cliffs, New Jersey 1973.

[471] J. Newman and K. E. Thomas-Alyea. Electrochemical Systems. 3rd edition. John Wiley & Sons Inc., Hoboken, New Jersey. 2004.

[472] N. M. Newmark. "A Method of Computation for Structural Dynamics". *ASCE Journal of the Engineering Mechanics Division*. Volume 85, Issue 3. p.67–94. 1959.

[473] T. V. Nguyen. "Modeling Two-Phase Flow in the Porous Electrodes of Proton Exchange Membrane Fuel Cells Using the Interdigitated Flow Fields". *Tutorials in Electrochemical Engineering Mathematical Modeling*. 99(14). 222–241. 1999.

[474] P. Nguyen, L. Vervisch, V. Subramanian and P. Domingo. "Multi-Dimensional Flamelet-Generated Manifolds for Partially Premixed Combustion". *Combustion and Flame*. 157. 43–61. 2010.

[475] F. Nicoud and F. Ducros. "Subgrid-Scale Stress Modelling Based on the Square of the Velocity Gradient Tensor.Flow". *Turbulence, and Combustion*. 62(3). 183–200. 1999.

[476] L. Nokleberg and T. Sontvedt. "Erosion of Oil and Gas Industry Choke Valves Using Computational Fluid Dynamics and Experiment". *International Journal of Heat and Fluid Flow*. 636–643. 1998.

[477] P. A. Nooren, H. A. Wouters, T. W. J. Peeters, D. Roekaerts, U. Maas, and D. Schmidt. "Monte carlo pdf modeling of a turbulent natural-gas diffusion flame". *Combustion Theory and Modeling*. 1. 79–96. 1997.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

[478] J. Norman, M. Porkashanian, and A. Williams. "Modelling the Formation and Emission of Environmentally Unfriendly Coal Species in Some Gasification Processes". *Fuel*. 76(13). 1201–1216. October 1997.

[479] W. H. Nurick. "Orifice Cavitation and Its Effects on Spray Mixing". *Journal of Fluids Engineering*. 98. 1976.

[480] R. Ocone, S. Sundaresan, and R. Jackson. "Gas-particle flow in a duct of arbitrary inclination with particle-particle interaction". *AIChE J*. 39. 1261–1271. 1993.

[481] B. Oesterle and T. Bui Dinh. "Experiments on the lift of a spinning sphere in a range of intermediate Reynolds numbers". *Exp. Fluids*. 25. 16-22. 1998.

[482] S. Ogawa, A. Umemura, and N. Oshima. "On the Equation of Fully Fluidized Granular Materials". *J. Appl. Math. Phys*. 31. 483. 1980.

[483] Y. I. Oka, T. Yoshida. "Practical Estimation of Erosion Damage Caused by Solid Particle Impact, Part 2: Mechanical Properties of Materials Directly Associated With Erosion Damage". *15th International Conference on Wear of Materials*. 259. 102-109. 2005.

[484] Y. Ohzawa, Y. Nagano, and T. Matsuo. "Fundamental Equations of Dry Spinning with an Example Calculation". *Proceedings, 5th Intern. Congress on Rheology*. 393–408. Kyoto: 1968.

[485] E. Olsson, G. Kreiss and S. Zahedi. "A Conservative Level Set Method for Two Phase Flow II". *J. Comput. Phys*. 225. 785–807. 2007.

[486] O. O. Oluwole, H.-W. Wong, R. C. Miake-Lye, and W. H. Green Jr. "Dynamic Spatial and Temporal Allocation of Reduced Chemical Kinetics Models in Combustion Computational Fluid Dynamics Simulations". *46th AIAA/ASME/SAE/ASEE Joint Propulsion Conference*. AIAA 2010-6827, 2010.

[487] S. Ookawara, M. Agrawal, D. Street, and K. Ogawa. "Modeling the motion of a sphere falling in a quiescent Newtonian liquid in a cylindrical tube by using the macroscopic particle model". *The Seventh World Congress of Chemical Engineering*. Glasgow, Scotland: C39-004. 2005.

[488] S. Ookawara, M. Agrawal, D. Street, and K. Ogawa. "Quasi-direct numerical simulation of lift force-induced particle separation in a curved microchannel by use of a macroscopic particle model". *Chemical Engineering Science*. 62. 9. 2454–2465. 2007.

[489] S.S. Ozarkar, X. Yan, S. Wang, C.C. Milioli, F.E. Milioli, S. Sundaresan. "Validation of filtered two-fluid models for gas–particle flows against experimental data from bubbling fluidized bed". *Powder Technology*. 284. 159–169. 2015.

[490] P. J. O'Rourke. "Collective Drop Effects on Vaporizing Liquid Sprays". *PhD thesis*. Princeton University, Princeton, New Jersey. 1981.

[491] P. J. O'Rourke and A. A. Amsden. "The TAB Method for Numerical Calculation of Spray Droplet Breakup". *SAE Technical Paper 872089*. SAE. 1987.

[492] P. J. O'Rourke and A. A. Amsden. "A Particle Numerical Model for Wall Film Dynamics in Port-Fuel Injected Engines". *SAE Paper 961961*. 1996.

[493] P. J. O'Rourke and A. A. Amsden. "A Spray/Wall Interaction Submodel for the KIVA-3 Wall Film Model". *SAE Paper 2000-01-0271*. 2000.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

[494] P. J. O'Rourke and F. V. Bracco. "Two Scaling Transformations for the Numerical Computation of Multidimensional Unsteady Laminar Flames". *Journal of Computational Physics*. 33 (2). 185 –203. 1979.

[495] O. Rybdylova et al.. "A model for multi-component droplet heating and evaporation and its implementation into ANSYS Fluent". *Int. Comm. Heat and Mass Transfer* . Vol. 90. 29–33. 2018.

[496] S. A. Orszag, V. Yakhot, W. S. Flannery, F. Boysan, D. Choudhury, J. Maruzewski, and B. Patel. "Renormalization Group Modeling and Turbulence Simulations". *In International Conference on Near-Wall Turbulent Flows, Tempe, Arizona*. 1993.

[497] S. Osher and J. A. Sethian,. "Fronts Propagating with Curvature-dependent Speed: Algorithms Based on Hamilton-Jacobi Formulations". *J. Comput. Phys*. 79. 12–49. 1988.

[498] M. Ostberg and K. Dam-Johansen. "Empirical Modeling of the Selective Non-Catalytic Reduction of NO Comparison with Large-Scale Experiments and Detailed Kinetic Modeling". *Chem. Engineering Science*. 49(12). 1897–1904. 1994.

[499] H. Ounis, G. Ahmadi, and J. B. McLaughlin. "Brownian Diffusion of Submicrometer Particles in the Viscous Sublayer". *Journal of Colloid and Interface Science*. 143(1). 66–277. 1991.

[500] M. N. Ozisik. *Radiative Transfer and Interactions with Conduction and Convection"*. Wiley, New York. 1973.

[501] S. A. Pandya, S. Venkateswaran, and T. H. Pulliam. "Implementation of dual-time procedures in overflow". *Technical Report AIAA-2003-0072*. American Institute of Aeronautics and Astronautics. 2003.

[502] J. C. Park, M. H. Kim and H. Miyata. "Fully Non-linear Free-surface Simulations By a 3D Viscous Numerical Wave Tank". *International Journal for Numerical Methods in Fluids*. 29. 685–703. 1999.

[503] U. Pasaogullari and C-Y. Wang. “Two-Phase transport and the role of micro-porous layer in polymer electrolyte fuel cells”. *Electrochimica Acta*. 49(25). 4359-4369. 2004.

[504] S. V. Patankar. *Numerical Heat Transfer and Fluid Flow*. Hemisphere, Washington, DC. 1980.

[505] S. V. Patankar. *Numerical Heat Transfer and Fluid Flow*. McGraw-Hill. Washington, New York, London 1980.

[506] M. A. Patterson and R. D. Reitz. "Modeling the Effects of Fuel Spray Characteristics on Diesel Engine Combustion and Emission". *SAE Paper*. SAE. 1998.

[507] D. Y. Peng and D. B. Robinson. "A New Two-Constant Equation of State". *Industrial and Engineering Chemistry: Fundamentals*. 15. 59–64. 1976.

[508] R. H. Perry, D. W. Gree, and J. O. Maloney. *Perry's Chemical Engineers' Handbook*. McGraw-Hill, New York. 6th edition. 1984.

[509] N. Peters. "Laminar Diffusion Flamelet Models in Non Premixed Combustion". *Prog. Energy Combust. Sci*. 10. 319–339. 1984.

[510] N. Peters. "Laminar Flamelet Concepts in Turbulent Combustion". *In 21st Symp. (Int'l.) on Combustion*. The Combustion Institute. 1231–1250. 1986.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

[511] N. Peters. *Turbulent Combustion*. Cambridge University Press, Cambridge, England. 2000.

[512] N. Peters and S. Donnerhack. *In 18th Symp. (Int'l.) on Combustion*. The Combustion Institute. 33. 1981.

[513] N. Peters and B. Rogg. "Reduced Kinetic Mechanisms for Applications in Combustion Systems". *In Lecture Notes in Physics*. m15. Springer-Verlag. 1992.

[514] M. Pilch and C. A. Erdman. "Discrete Element Method (DEM) Simulation and Processing of Mo/AL2O3 Granules in Fluidizing Bed". *Proc. Nat. Sci. Counc. RAOC(A)*. 24. 5. 394-404. 2000.

[515] M. Pilch and C. A. Erdman. "Use of Breakup Time Data and Velocity History Data to Predict the Maximum Size of Stable Fragments for Acceleration-Induced Breakup of a Liquid Drop". *Int. J. Multiphase Flow*. 13. 6. 741–757. 1987.

[516] K. K. Pillai. "The Influence of Coal Type on Devolatilization and Combustion in Fluidized Beds". *J. Inst. Energy*. 142. 1981.

[517] U. Piomelli, P. Moin and J. H. Ferziger. "Model Consistency in Large-Eddy Simulation of Turbulent Channel Flow". *Physics of Fluids*. 31. 1884–1894. 1988.

[518] H. Pitsch. "A Consistent Level Set Formulation For Large-Eddy Simulation of Premixed Turbulent Combustion". *Combustion and Flame*. 143. 587–598. 2005.

[519] H. Pitsch, H. Barths, and N. Peters. "Three-Dimensional Modeling of NOx and Soot Formation in DI-Diesel Engines Using Detailed Chemistry Based on the Interactive Flamelet Approach". *SAE Paper 962057*. SAE. 1996.

[520] H. Pitsch and N. Peters. "A Consistent Flamelet Formulation for Non-Premixed Combustion Considering Differential Diffusion Effects". *Combustion and Flame*. 114. 26–40. 1998.

[521] T. Poinsot and D. Veynante. *Theoretical and Numerical Combustion*. McGraw-Hill, New York. 2001.

[522] B. E. Poling, J. M. Prausnitz, and J. P. OConnell. *The Properties of Gases and Liquids*. McGraw-Hill, New York, 5th edition. 2001.

[523] S. B. Pope. "Computationally efficient implementation of combustion chemistry using in-situ adaptive tabulation". *Combustion Theory and Modeling*. 1. 41–63. 1997.

[524] S. B. Pope. "Gibbs Function Continuation for the Stable Computation of Chemical Equilibrium". *Combustion and Flame*. 139. 222–226. 2004.

[525] S. B. Pope. "Pdf methods for turbulent reactive flows". *Progress Energy Combustion Science*. 11. 119. 1985.

[526] S. B. Pope. "Probability Distributions of Scalars in Turbulent Shear Flow". *Turbulent Shear Flows*. volume 2. pages 7–16. 1979.

[527] S. B. Pope. *Turbulent Flows*. Cambridge University Press, Cambridge, England. 2000.

[528] B. Popoff and M. Braun. "A Lagrangian Approach to Dense Particulate Flows". *In International Conference on Multiphase Flow*. Leipzig, Germany. 2007.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

[529] C. Prakash. “Two phase model for binary liquid-solid phase change: Parts I and II”. *Numerical Heat Transfer*. 18. 2. 131–167. 1990.

[530] L. Prandtl. "Uber die ausgebildete Turbulenz". ZAMM. 5. 136–139. 1952.

[531] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *Numerical Recipes*. Cambridge University Press. Cambridge, England,1992.

[532] M. Prince and H. Blanch. *"Bubble Coalescence and Break-Up in Air-Sparged Bubble Columns". AIChE Journal*. 36. 10. 1485–1499. 1990.

[533] I. Proudman. "The Generation of Noise by Isotropic Turbulence". *Proc. Roy. Soc*. A214. 119. 1952.

[534] A. Rabbani and M. Rokni . "Effect of nitrogen crossover on purging strategy in PEM fuel cell systems". *Applied Energy*. 111. 1061–1070. 2013.

[535] G. D. Raithby and E. H. Chui. "A Finite-Volume Method for Predicting a Radiant Heat Transfer in Enclosures with Participating Media". *J. Heat Transfer*. 112. 415–423. 1990.

[536] D. Ramkrishna. *Population Balances Theory and Applications to Particulate Systems in Engineering*. Academic Press. San Diego, CA,2000.

[537] A. D. Randolph and M. A. Larson. Theory of Particulate Processes. *Analysis and Techniques of Continuous Crystallization*. Academic Press. San Diego, CA1971.

[538] W. E. Ranz. "Some Experiments on Orifice Sprays". *Canadian Journal of Chemical Engineering*. 175. 1958.

[539] W. E. Ranz and W. R. Marshall, Jr. "Vaporation from Drops, Part I". *Chem. Eng. Prog*. 48(3). 141–146. March 1952.

[540] W. E. Ranz and W. R. Marshall, Jr. "Evaporation from Drops, Part I and Part II". *Chem. Eng. Prog*. 48(4). 173–180. April 1952.

[541] A. V. Rao. *Dynamics of Particles and Rigid Bodies. A Systematic Approach*. Cambridge University Press. Cambridge, England 2006.

[542] R. D. Rauch, J. T. Batira, and N. T. Y. Yang. "Spatial Adaption Procedures on Unstructured Meshes for Accurate Unsteady Aerodynamic Flow Computations". *Technical Report AIAA-91-1106*. AIAA. 1991.

[543] Reaction Design. http://www.reactiondesign.com . 2012.

[544] R. D. Reitz. "Mechanisms of Atomization Processes in High-Pressure Vaporizing Sprays". *Atomization and Spray Technology*. 3. 309–337. 1987.

[545] R. D. Reitz and F. V. Bracco. "Mechanism of Atomization of a Liquid Jet". *Phys. Fluids*. 26(10). 1982.

[546] R. D. Reitz and F. V. Bracco. "Mechanisms of Breakup of Round Liquid Jets". *The Encyclopedia of Fluid Mechanics, ed. N. Cheremisnoff*. 3. 223–249. 1986.

[547] C. H. Rexroth, H. J. Bauer, and S. Wittig. “DISC—An efficient method for the discretization of convection on unstructured grids”. Aerospace Science and Technology. 6. 1997.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

[548] W. C. Reynolds. "Thermodynamic Properties in SI: Graphs, Tables, and Computational Equations for 40 Substances". Department of Mechanical Engineering, Stanford University. 1979.

[549] W. C. Reynolds. "Fundamentals of turbulence for turbulence modeling and simulation". *Lecture Notes for Von Karman Institute Agard Report No. 755*. 1987.

[550] M. Renksizbulut and M. C. Yuen. "Experimental Study of Droplet Evaporation in a High-Temperature Air Stream". *Transactions of the ASME*. 105 (2). 384-388. 1983.

[551] C. M. Rhie and W. L. Chow. "Numerical Study of the Turbulent Flow Past an Airfoil with Trailing Edge Separation". *AIAA Journal*. 21(11). 1525–1532. November 1983.

[552] H. S. Ribner. "The Generation of Sound by turbulent jets". *In Advances in Applied Mathematics*. Academic, New York. 1964.

[553] S. Richard, O. Colin, O. Vermorel, A. Benkenida, C. Angelberger, and D. Veynante. "Towards Large Eddy Simulation of Combustion in Spark Ignition Engines". *Proceedings of the Combustion Institute*. 31. 2. 3059–3066. 2007.

[554] J. R. Richardson and W. N. Zaki. "Sedimentation and Fluidization: Part I". *Trans. Inst. Chem. Eng*. 32. 35–53. 1954.

[555] C. E. Roberts and R. D. Matthews. "Development and Application of an Improved Ring Pack Model for Hydrocarbon Emissions Studies". *SAE Technical Paper 961966*. SAE. 1996.

[556] P. L. Roe. "Characteristic based schemes for the Euler equations". *Annual Review of Fluid Mechanics*. 18. 337–365. 1986.

[557] R. Rota, D. Antos, E. F. Zanoelo, and M. Morbidelli. "Experimental and Modeling Analysis of the NOxOUT Process". *Chemical Engineering Science*. 57. 27–38. 2002.

[558] S. I. Rubinow and J. B. Keller. "The transverse force on a spinning sphere moving in a viscous liquid". *J. Fluid Mech.*. 11. 447-459. 1961.

[559] P. G. Saffman. "The Lift on a Small Sphere in a Slow Shear Flow". *J.Fluid Mech*. 22. 385–400. 1965.

[560] P. G. Saffman. Corrigendum to: "The Lift on a Small Sphere in a Slow Shear Flow". *J.Fluid Mech*. 31. 624. 1968.

[561] P. G. Saffman and J. S. Turner. On the Collision of Droplets in Turbulent Clouds. *Journal of Fluid Mechanics,*. 1. 16–30. 1956.

[562] A. Sarkar, F. E. Milioli, S. Ozarkar, T. Li, X. Sun, S. Sundaresan. "Filtered sub-grid constitutive models for fluidized gas-particle flows constructed from 3-D simulations". *Chemical Engineering Science*. 152. 443–456. 2016.

[563] S. Sarkar and L. Balakrishnan. "Application of a Reynolds-Stress Turbulence Model to the Compressible Shear Layer". ICASE Report 90-18*NASA CR 182002*. 1990.

[564] S. Sarkar and M. Y. Hussaini. "Computation of the sound generated by isotropic turbulence". *NASA Contract Report 93-74*. NASA Langley Research Center, Hampton, VA. 1993.

[565] M. M. Salama and E. S. Venkatesh. "Evaluation of api rp14e erosional velocity limitations for offshore gas wells". *In OTC Conference, Houston, TX*. 371–376. May 1983.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

[566] R. Sander. Compilation of Henry's Law Constants for Inorganic and Organic Species of Potential Importance in Environmental Chemistry. http://www.henrys-law.org/:

[567] G.S.R. Sarma. "Physico–chemical Modelling in Hypersonic Flow Simulation". *Progress in Aerospace Sciences*. 36. 3-4. 281-349. 2000.

[568] Y. Sato and K. Sekoguchi. "Liquid Velocity Distribution in Two-Phase Bubbly Flow". *Int. J. Multiphase Flow*. 2. 79. 1979.

[569] S. S. Sazhin. "An Approximation for the Absorption Coefficient of Soot in a Radiating Gas". *Manuscript*. Fluent Europe, Ltd. 1994.

[570] S. S. Sazhin. "Advanced Models of Fuel Droplet Heating and Evaporation". *Progress in Energy and Combustion Science*. Elsevier Science. 32. 162–214. 2006.

[571] L. C. Scalabrin. "Numerical Simulation of Weakly Ionized Hypersonic Flow over Reentry Capsules". *PhD thesis*. University of Michigan, Ann Arbor, Michigan. 2007.

[572] D. G. Schaeffer. "Instability in the Evolution Equations Describing Incompressible Granular Flow". *J. Diff. Eq*. 66. 19–50. 1987.

[573] R. W. Schefer, M. Namazian, and J. Kelly. *"In Combustion Research Facility News*. 3:4. Sandia. 1991.

[574] L. Schiller and Z. Naumann. *"Z. Ver. Deutsch. Ing*. 77. 318. 1935.

[575] H. Schlichting and K. Gersten. *Grenzschicht-Theorie*. Springer-Verlag, Berlin, Heidelberg, New York. 1997.

[576] H. Schlichting and K. Gersten. *Grenzschicht-Theorie*. Springer. Berlin, Heidelberg, New York, London 1997.

[577] R. Schmehl, G. Klose, G. Maier, S. Wittig. "Efficient Numerical Calculation of Evaporating Sprays in Combustion Chamber Flows". *In RTO AVTSymposium on Gas Turbine Engine Combustion Emissions and Alternative Fuels, Lisbon, Portugal, 1998*. RTO MP-14. 51.1–51.14. 1999.

[578] D. P. Schmidt and M. L. Corradini. "Analytical Prediction of the Exit Flow of Cavitating Orifices". *Atomization and Sprays*. 7. 6. 1997.

[579] D. P. Schmidt, M. L. Corradini, and C. J. Rutland. "A Two-Dimensional, Non-Equilibrium Model of Flashing Nozzle Flow". *In 3rd ASME/JSME Joint Fluids Engineering Conference*. 1999.

[580] D. P. Schmidt, I. Nouar, P. K. Senecal, C. J. Rutland, J. K. Martin, and R. D. Reitz. "Pressure-Swirl Atomization in the Near Field". *SAE Paper 01-0496*. SAE. 1999.

[581] M. Schmoluchowski. Versuch Einen Mathematischen Theorie der Koagulationstechnik Kolloider Lösungen. *Zeitschrift für Physikalische Chemie*. 92. 129–168. 1917.

[582] G. H. Schnerr and J. Sauer. "Physical and Numerical Modeling of Unsteady Cavitation Dynamics". *In Fourth International Conference on Multiphase Flow, New Orleans, USA*. 2001.

[583] H. Scholz. "Modellierung und Untersuchung des Wärme- und Stofftransports und von Flutungsphänomenen in Niedertemperatur-PEM-Brennstoffzellen". *PhD Thesis*. 2015.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

[584] P. K. Senecal, D. P. Schmidt, I. Nouar, C. J. Rutland, and R. D. Reitz. "Modeling High Speed Viscous Liquid Sheet Atomization". *International Journal of Multiphase Flow*. 25(6–7). 1073–1097. 1999.

[585] E. Sergent. "Vers une methodologie de couplage entre la Simulation des Grandes Echelles et les modeles statistiques". *PhD thesis*. L'Ecole Centrale de Lyon, Lyon, France. 2002.

[586] N. G. Shah. "A New Method of Computation of Radiant Heat Transfer in Combustion Chambers". *PhD thesis*. Imperial College of Science and Technology, London, England. 1979.

[587] D.R. Shaver, M.Z. Podowski. "Modeling of interfacial forces for bubbly flows in subcooled boiling conditions". *Transactions of the American Nuclear Society (Proc. of ANS Winter Meeting)*. 113(1). 1368–1371. 2015.

[588] T.-H. Shih, W. W. Liou, A. Shabbir, Z. Yang, and J. Zhu. "A New k- ε Eddy-Viscosity Model for High Reynolds Number Turbulent Flows - Model Development and Validation". *Computers Fluids*. 24(3). 227–238. 1995.

[589] M. L. Shur, P. R. Spalart, M. K. Strelets, and A. K. Travin. "A Hybrid RANS-LES Approach With Delayed-DES and Wall-Modelled LES Capabilities". *International Journal of Heat and Fluid Flow*. 29:6. December 2008. 1638-1649.

[590] M. Shur, P. R. Spalart, M. Strelets, and A. Travin. "Detached-Eddy Simulation of an Airfoil at High Angle of Attack". *In 4th Int. Symposium on Eng. Turb. Modeling and Experiments, Corsica, France*. May 1999.

[591] Michael L. Shur, Philippe R. Spalart, Michael K. Strelets, Andrey K. Travin; Synthetic Turbulence Generators for RANS-LES Interfaces in Zonal Simulations of Aerodynamic and Aeroacoustic Problems; Flow Turbulence Combust (2014) 93: 63. .

[592] M. L. Shur, M. K. Strelets, A. K. Travin and P. R. Spalart. "Turbulence Modeling in Rotating and Curved Channels: Assessing the Spalart-Shur Correction". *AIAA Journal*. 38(5). 2000. 2000.

[593] R. Siegel and J. R. Howell. *"Thermal Radiation Heat Transfer*. Hemisphere Publishing Corporation, Washington DC. 1992.

[594] R. Siegel and C. M. Spuckler. "Effect of Refractive Index and Diffuse or Specular Boundaries on a Radiating Isothermal Layer". *J. Heat Transfer*. 116. 787–790. 1994.

[595] O. Simonin and P. L. Viollet. "Modeling of Turbulent Two-Phase Jets Loaded with Discrete Particles". *Phenomena in Multiphase Flows*. 259–269. 1990.

[596] O. Simonin and P. L. Viollet. "Predictions of an Oxygen Droplet Pulverization in a Compressible Subsonic Coflowing Hydrogen Flow". *Numerical Methods for Multiphase Flows*. FED91. 65–82. 1990.

[597] A. K. Singhal, H. Y. Li, M. M. Athavale, and Y. Jiang. "Mathematical Basis and Validation of the Full Cavitation Model". *ASME FEDSM'01*. New Orleans, Louisiana 2001.

[598] Y. R. Sivathanu and G. M. Faeth. "Generalized State Relationships for Scalar Properties in Non-Premixed Hydrocarbon/AirFlames". *Combustion and Flame*. 82. 211–230. 1990.

[599] S. Skogestad. Chemical and energy process engineering. Boca Raton, FLCRC Press. 2009.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

[600] J. Smagorinsky. "General Circulation Experiments with the Primitive Equations. I. The Basic Experiment". *Month. Wea. Rev*. 91. 99–164. 1963.

[601] P. E. Smirnov and F. R. Menter. "Sensitization of the SST Turbulence Model to Rotation and Curvature by Applying the Spalart-Shur Correction Term". *ASME Paper GT 2008-50480*. Berlin, Germany. 2008.

[602] R. Smirnov, S. Shi, and I. Celik. "Random Flow Generation Technique for Large Eddy Simulations and Particle-Dynamics Modeling". *Journal of Fluids Engineering*. 123. 359–371. 2001.

[603] H. Smith, M. Zöchbauer, and T. Lauer. "Advanced Spray Impingement Modelling for an Improved Prediction Accuracy of the Ammonia Homogenisation in SCR Systems". *"SAE Technical Paper 2015-01-1054"*. doi:10.4271/2015-01-1054. 2015.

[604] I. W. Smith. *"Combustion and Flame*. 17. 421. 1971.

[605] I. W. Smith. "The Intrinsic Reactivity of Carbons to Oxygen". *Fuel*. 57. 409–414. 1978.

[606] I. W. Smith. "The Combustion Rates of Coal Chars: A Review". *In 19th Symp. (Int'l.) on Combustion*. The Combustion Institute. 1045–1065. 1982.

[607] J. M. Smith and H. C. Van Ness. *Introduction to Chemical Engineering Thermodynamics"*. McGraw Hill, New Jersey. 1986.

[608] T. F. Smith, Z. F. Shen, and J. N. Friedman. "Evaluation of Coefficients for the Weighted Sum of Gray Gases Model". *J. Heat Transfer*. 104. 602–608. 1982.

[609] K. Smith and C. Y. Wang. "Solid-state Diffusion Limitations on Pulse Operation of a Lithium ion Cell for Hybrid Electric Vehicles". *Journal of Power Sources*. 161. 628-639. 2006.

[610] L. D. Smoot and P. J. Smith. "NOx Pollutant Formation in a Turbulent Coal System". *In Coal Combustion and Gasification*. Plenum, Plenum, NY373. 1985.

[611] D. O. Snyder, E. K. Koutsavdis, and J. S. R. Anttonen. "Transonic store separation using unstructured CFD with dynamic meshing". *Technical Report AIAA-2003-3913*. 33rd AIAA Fluid Dynamics Conference and Exhibit American Institute of Aeronautics and Astronautics. 2003.

[612] A. Sokolichen, G. Eigenberger, and A. Lapin. "Simulation of Buoyancy Driver Bubbly Flow: Established Simplifications and Open Questions". *Journal Review, AIChE Journal*. 50:1. 24–45. January 2004.

[613] M. S. Solum, R. J. Pugmire, and D. M. Grant. *"Energy and Fuels"*. 3. 187. 1989.

[614] M. Sommerfeld, and N. Huber. "Experimental analysis and modeling of particle-wall collisions". *Int. J. Multiphase Flow*. 25. 1457—1489. 1999.

[615] C. Soteriou, R. Andrews, and M. Smith. "Direct Injection Diesel Sprays and the Effect of Cavitation and Hydraulic Flip on Atomization". *SAE Paper 950080*. SAE. 1995.

[616] P. R. Spalart. "Strategies for Turbulence Modelling and Simulations". *International Journal of Heat and Fluid Flow*. 21. 3. 252–263. 2000.

[617] P. R. Spalart. "Young-person's guide to detached-eddy simulation grids". *NASA NASA/CR-2001-211032*. NASA Langley Research Center. 2001.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

[618] P. Spalart and S. Allmaras. "A one-equation turbulence model for aerodynamic flows". *Technical Report AIAA-92-0439*. American Institute of Aeronautics and Astronautics. 1992.

[619] P. R. Spalart, S. Deck, M. L. Shur, K. D. Squires, M. K. Strelets, and A. Travin. "A new version of detached-eddy simulation, resistant to ambiguous grid densities". *Theoretical and Computational Fluid Dynamics*. 20. 181–195. 2006.

[620] P. R. Spalart and M. L. Shur. "On the Sensitization of Turbulence Models to Rotation and Curvature". *Aerospace Sci. Tech*. 1(5). 297–302. 1997.

[621] D. B. Spalding. "Mixing and chemical reaction in steady confined turbulent flames". *In 13th Symp. (Int'l.) on Combustion*. The Combustion Institute. 1970.

[622] C. G. Speziale, S. Sarkar, and T. B. Gatski. "Modelling the Pressure-Strain Correlation of Turbulence: An Invariant Dynamical Systems Approach". *J. Fluid Mech*. 227. 245–272. 1991.

[623] T. E. Springer, T. A. Zawodzinski, and S. Gottesfeld. "Polymer Electrolyte Fuel Cell Model". *J. Electrochemical Soc*. 138:8. August 1991.

[624] T. E. Springer, T. A. Zawodzinski, and S. Gottesfeld. "Polymer Electrolyte Fuel Cell Model". *Journal, Electrochemical Society*. 138. 2334–2342. 1991.

[625] H. B. Squire. "Investigation of the Instability of a Moving Liquid Film". *British Journal of Applied Physics*. 4. 167. 1953.

[626] D. W. Stanton and C. J. Rutland. "Modeling Fuel Film Formation and Wall Interaction in Diesel Engines". *SAE Paper 960628*. 1996.

[627] D. W. Stanton and C. J. Rutland. "Multi-Dimentional Modeling of Heat and Mass Transfer of Fuel Films Resulting from Impinging Sprays". *SAE 980132*. 1998.

[628] D. W. Stanton and C. J. Rutland. "Multi-Dimensional Modeling of Thin Liquid Films and Spray-Wall Interactions Resulting from Impinging Sprays". *International Journal of Heat and Mass Transfer*. 41. 3037–3054. 1998.

[629] R. C. Steele, P. C. Malte, D. G. Nichol, and J. C. Kramlich. "NOx and N_2O in Lean-Premixed Jet-Stirred Flames". *Combustion and Flame*. 100. 440–449. 1995.

[630] H. Steiner, A. Kobor and L. Gebhard. "A wall heat transfer model for subcooled boiling flow". *International Journal of Heat and Mass Transfer*. 48. 4161–4173. 2005.

[631] P. H. Steward, C. W. Larson, and D. Golden. *"Combustion and Flame*. 75. 1989.

[632] S. Subramaniam and S. B. Pope. "A Mixing Model for Turbulent Reactive Flows Based on Euclidean Minimum Spanning Trees". *Combustion and Flame*. 115. 487–514. 1998.

[633] K. Sutton and P. A. Gnoffo. "Multi-component Diffusion with Application to Computational Aerothermodynamics". *AIAA Paper 98-2575*. AIAA. 1998.

[634] M. Syamlal. "The Particle-Particle Drag Term in a Multiparticle Model of Fluidization". *National Technical Information Service, Springfield, VA*. DOE/MC/21353-2373, NTIS/DE87006500, 1987.

[635] M. Syamlal and T. O'Brien. "The Derivation of a Drag Coefficient Formula from Velocity-Voidage Correlations". 1987.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

[636] M. Syamlal and T. J. O'Brien. "Computer Simulation of Bubbles in a Fluidized Bed". *AIChE Symp. Series*. 85. 22–31. 1989.

[637] M. Syamlal, W. Rogers and T. J. O'Brien. *MFIX Documentation: Volume1, Theory Guide*>. National Technical Information Service, Springfield, VA. DOE/METC-9411004, NTIS/DE9400087, 1993.

[638] D. Tabacco, C. Innarella, and C. Bruno. "Theoretical and Numerical Investigation on Flameless Combustion". *Combustion Science and Technology*. 2002.

[639] W. Tabakoff and T. Wakeman. "Measured particle rebound characteristics useful for erosion prediction". *ASME paper 82-GT-170*. 1982.

[640] T. Takamasa and A. Tomiyama. . "Three-Dimensional Gas-Liquid Two-Phase Bubbly Flow in a C-Shaped Tube". *Ninth International Topical Meeting on Nuclear Reactor Thermal Hydraulics (NURETH-9)*. San Francisco, CA. 1999.

[641] L. Talbot et al. "Thermophoresis of Particles in a Heated Boundary Layer". *J. Fluid Mech*. 101(4). 737–758. 1980.

[642] I. Tanasawa. "Advances in Condensation Heat Transfer". *Advances in Heat Transfer*. 21. 55–139. 1991.

[643] L. Tay-Wo-Chong, A. Scarpato, and W. Polifke. "LES Combustion Model With Stretch and Heat Loss Effects for Prediction of Premix Flame Characteristics and Dynamics". *Proceedings of ASME Turbo Expo 2017: Turbomachinery Technical Conference and Exposition*. Paper No: GT2017-63357. 2017.

[644] G. I. Taylor. "The Shape and Acceleration of a Drop in a High Speed Air Stream, Technical Report". *In the Scientific Papers of G. I. Taylor*. ed., G. K. Batchelor. 1963.

[645] P. B. Taylor and P. J. Foster. "Some Gray Weighting Coefficients for CO_2-H_2O-Soot Mixtures". *Int. J. Heat Transfer*. 18. 1331–1332. 1974.

[646] R. Taylor and R. Krishna. *Multicomponent Mass Transfer*. Wiley, New York. 1993.

[647] A. Tentner, S. Lo and V. Kozlov. "Advances in Computational Fluid Dynamics Modeling of Two-Phase Flow in a Boiling Water Reactor Fuel Assembly". *In International Conference on Nuclear Engineering*. Miami, Florida. 2006.

[648] A. Teraji, T. Tsuda, T. Noda, M. Kubo, and T. Itoh. "Development of a Novel Flame Propagation Model (UCFM: Universal Coherent Flamelet Model) for SI Engines and Its Application to Knocking Prediction". *SAE 2005 World Congress & Exhibition*. 2005.

[649] P. A. Tesner, T. D. Snegiriova, and V. G. Knorre. "Kinetics of Dispersed Carbon Formation". *Combustion and Flame*. 17. 253–260. 1971.

[650] P. Thakre, R. Rawat, R. Clayton and V. Yang. "Mechanical Erosion of Graphite Nozzle in Solid-Propellant Rocket Motor". *J. of Propulsion and Power*. 29. 593-601. 2013.

[651] Tiedemann and Newman. S.Gross Editor. "Battery Design and Optimization". *Proceedings, Electrochemical Soc*. Princeton, NJ. 79-1. 39. 1979.

[652] V. I. Tolubinski and D. M. Kostanchuk. Vapor bubbles growth rate and heat transfer intensity at subcooled water boiling. *4th International Heat Transfer Conference, Paris, France*. 1970.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

[653] A. Tomiyama. "Struggle with computational bubble dynamics". *Third International Conference on Multiphase Flow, Lyon, France*. June 8–12, 1998.

[654] A. Tomiyama, H. Tarnai, I. Zun and S. Hosokama. "Transverse Migration of Single Bubbles in Simple Shear Flow". *Chem. Eng. Sci.* 57. 1849–1858. 2002.

[655] B. Toole-ONeil, S. J. Tewalt, R. B. Finkleman, and D. J. Akers. "Mercury Concentration in Coal Unravelling the Puzzle". *Fuel*. 78(1). 47–74,. January 1999.

[656] A. A. Troshko and Y. A. Hassan. "A Two-Equation Turbulence Model of Turbulent Bubbly Flow". *International Journal of Multiphase Flow*. 22(11). 1965–2000. 2001.

[657] Y. Tsuji, T. Oshima, and Y. Morikawa. "Numerical simulation on pneumatic conveying in horizontal pipe". *KONA-Powder Sci. Tech.Japan*. 3. 38-51. 1985.

[658] E. Turkel and V. N. Vatsa. "Choice of variables and preconditioning for time dependent problems". *Technical Report AIAA-2003-3692*. 16th AIAA Computational Fluid Dynamics Conference, Orlando, FloridaAmerican Institute of Aeronautics and Astronautics. June 2003.

[659] O. Ubbink. "Numerical Prediction of Two Fluid Systems With Sharp Interfaces". *PhD thesis*. Imperial College of Science, Technology and Medicine, London, England. 1997.

[660] Sukkee Um, C. Y. Wang, and K. S. Chen. “Computational Fluid Dynamics Modeling of Proton Exchange Membrane Fuel Cells”. *Journal, Electrochemical Society*. 147(12). 4485–4493. 2000.

[661] H. C. Unal. "Maximum Bubble Diameter, Maximum Bubble Growth Time and Bubble Growth Rate During Subcooled Nucleate Flow Boiling of Water Up To $17.7MN/m^2$". *Int. J. Heat Mass Transfer*. 19. 643–649. 1976.

[662] B. Van Leer. "Toward the Ultimate Conservative Difference Scheme. IV. A Second Order Sequel to Godunov's Method". *Journal of Computational Physics*. 32. 101–136. 1979.

[663] A. van Oijen and L. P. H. de Goey. "Modelling of Premixed Laminar Flames Using Flamelet-Generated Manifolds". *Combust. Sci. Tech*. 161. 113–137. 2000.

[664] J. P. Vandoormaal and G. D. Raithby. "Enhancements of the SIMPLE Method for Predicting Incompressible Fluid Flows". *Numer.Heat Transfer*. 7. 147–163. 1984.

[665] S. A. Vasquez and V. A. Ivanov. "A Phase Coupled Method for Solving Multiphase Problems on Unstructured Meshes". *In Proceedings of ASME FEDSM'00*. ASME 2000 Fluids Engineering Division Summer Meeting, Boston. June 2000.

[666] V. Venkatakrishnan. "On The Accuracy of Limiters and Convergence to Steady State Solutions". *Technical Report AIAA-93-0880*. American Institute of Aeronautics and Astronautics. January 1993.

[667] S. Venkateswaran, J. M. Weiss, and C. L. Merkle. "Propulsion Related Flowfields Using the Preconditioned Navier-Stokes Equations". *Technical Report AIAA-92-3437*. AIAA/ASME/SAE/ASEE 28th Joint Propulsion Conference, Nashville, TN. July 1992.

[668] O. Vermorel, S. Richard, O. Colin, C. Angelberger, A. Benkenida, and D. Veynante. "Towards the Understanding of Cyclic Variability in a Spark Ignited Engine Using Multi-Cycle LES". *Combustion and Flame*. 156. 8. 1525–1541. 2009.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

[669] D. Veynante and L. Vervisch. "Turbulent Combustion Modeling". *Progress in Energy and Combustion Science*. Elsevier Science. 28. 3. 193–266. 2002.

[670] J. R. Viegas, M. W. Rubesin, and C. C. Horstman. "On the Use of Wall Functions as Boundary Conditions for Two-Dimensional Separated Compressible Flows". *Technical Report AIAA-85-0180*. AIAA 23rd Aerospace Sciences Meeting, Reno, Nevada1985.

[671] R. D. Vigil and R. M. Ziff. On the Stability of Coagulation–Fragmentation Population Balances. *Journal of Colloids and Interface Science*. 133(1). 257–264. 1989.

[672] K. Viswanathan. "Aeroacoustics of hot jets". *Journal of Fluid Mechanics*. 516. 39–82. 2004.

[673] V. R. Voller. "Modeling Solidification Processes". *Technical report*. Mathematical Modeling of Metals Processing Operations Conference, Palm Desert, CAAmerican Metallurgical Society. 1987.

[674] V. R. Voller, A. D. Brent, and C. Prakash. "The Modeling of Heat, Mass and Solute Transport in Solidification Systems". *Int. J. Heat Mass Transfer*. 32(9). 1719–1731. 1989.

[675] V. R. Voller, A. D. Brent, and K. J. Reid. "A Computational Modeling Framework for the Analysis of Metallurgical Solidification Process and Phenomena". *Technical report*. Conference for Solidification ProcessingRanmoor House, Sheffield, England. September 1987.

[676] V. R. Voller and C. Prakash. "A Fixed-Grid Numerical Modeling Methodology for Convection-Diffusion Mushy Region Phase-Change Problems". *Int. J. Heat Mass Transfer*. 30. 1709–1720. 1987.

[677] V. R. Voller and C. R. Swaminathan. "Generalized Source-Based Method for Solidification Phase Change". *Numer. Heat Transfer B,*. 19(2). 175–189. 1991.

[678] K. S. Vorres. "User's handbook for the Argonne premium coal sample bank". Argonne National Laboratory, supported by DOE contract W-31-109-ENG-38, September 1989. Also K. S. Vorres, ACS Div. Fuel Chem. preprint 32. 4. 1987.

[679] M. P. Vukalovitch. Thermodynamic Properties of Steam. *6th edition*. 1958. (Mashgis, Moscow).

[680] D. Wadnerkar, M. Agrawal, and V. Pareek. "Terminal Velocity of Particles Falling in Non-Newtonian Yield Pseudo-plastic Fluids using a Macroscopic Particle Model". *In 7th International Conference on Multiphase FLOW*. ICMF 2010Tampa, FL: 2010.

[681] S. Wallin and A. Johansson. "A complete explicit algebraic Reynolds stress model for incompressible and compressible flows". *Journal of Fluid Mechanics*. 403. 89–132. 2000.

[682] D. Keith Walters and Davor Cokljat. "A three-equation eddy-viscosity model for reynolds-averaged navier-stokes simulations of transitional flows". *Journal of Fluids Engineering*. 130. December 2008.

[683] D. K. Walters and N. M. Wolgemuth. "A new interface-capturing discretization for numerical solution of the volume fraction equation in two-phase flows". *International Journal for Numerical Methods in Fluids*. 60. 893-918. 2009.

[684] G. Wang, M. Boileau, and D. Veynante. "Implementation of a dynamic thickened flame model for large eddy simulations of turbulent premixed combustion". *Combustion and Flame*. 158. 11. 2199–2213. 2011.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

[685] L. P. Wang. "On the Dispersion of Heavy Particles by Turbulent Motion". *PhD thesis*. Washington State University. 1990.

[686] T. Wang, J. Wang, Y. Jin. "Theoretical prediction of flow regime transition in bubble columns by the population balance model". *Che. Eng. Sci.*. 60. 22. 6199–6209. 2005.

[687] W. Wang and J. Li. “Simulation of gas-solid two-phase flow by a multi-scale CFD approach - Extension of the EMMS model to the sub-grid level”. *Chemical Engineering Science*. 62. 208–231. 2007.

[688] Yun Wang and C. Y. Wang. “Transient analysis of polymer electrolyte fuel cells”. *Electrochimica Acta*. 50. 1307-1315. 2005.

[689] Z. J. Wang. "A Fast Nested Multi-grid Viscous Flow Solver for Adaptive Cartesian/Quad Grids". *International Journal for Numerical Methods in Fluids*. 33. 657–680. 2000.

[690] J. Warnatz. "NOx Formation in High Temperature Processes". University of Stuttgart, Germany.

[691] G. P. Warren, W. K. Anderson, J. L. Thomas, and S. L. Krist. "Grid convergence for adaptive methods". *Technical Report AIAA-91-1592*. AIAA10th Computational Fluid Dynamics Conference, Honolulu, HawaiiAmerican Institute of Aeronautics and Astronautics. June 1991.

[692] C. Weber. "Zum Zerfall eines Flüssigkeitsstrahles". *ZAMM*. 11. 136–154. 1931.

[693] J. M. Weiss, J. P. Maruszewski, and W. A. Smith. "Implicit Solution of the Navier-Stokes Equations on Unstructured Meshes". *Technical Report AIAA-97-2103*. 13th AIAA CFD Conference, Snowmass, COJuly 1997.

[694] J. M. Weiss, J. P. Maruszewski, and W. A. Smith. "Implicit Solution of Preconditioned Navier-Stokes Equations Using Algebraic Multigrid". *AIAA Journal*. 37(1). 29–36. 1999.

[695] J. M. Weiss and W. A. Smith. "Preconditioning Applied to Variable and Constant Density Flows". *AIAA Journal*. 33(11). 2050–2057. November 1995.

[696] R. M. Wellek, A. K. Agrawal, and A. H. P. Skelland. "Shape of liquid drops moving in liquid media". *AIChE J.*. 12(5). 854–862. September 1966.

[697] C.-Y. Wen and Y. H. Yu. "Mechanics of Fluidization". *Chem. Eng. Prog. Symp. Series*. 62. 100–111. 1966.

[698] Z. Wen, S. Yun, M. J. Thomson, and M. F. Lightstone. "Modeling Soot Formation in Turbulent Kerosene/Air Jet Diffusion Flames". *Combustion and Flame*. 135. 323–340. 2003.

[699] H. Werner and H. Wengle. "Large-Eddy Simulation of Turbulent Flow Over and Around a Cube in a Plate Channel". *In Eighth Symposium on Turbulent Shear Flows, Munich, Germany*. 1991.

[700] C. Westbrook and F. Dryer. "Chemical Kinetic Modelling of Hydrocarbon Combustion". *Prog. Energy Comb. Sci*. 1. 1984.

[701] A. A. Westenberg. *"Comb. Sci. Tech"*. 4. 59. 1971.

[702] J.C. Wheeler. "Modified moments and Gaussian quadratures". *Rocky Mount. J.* 4 (2). 287–296. 1974.

[703] F. White and G. Christoph. "A Simple New Analysis of Compressible Turbulent Skin Friction Under Arbitrary Conditions". *Technical Report AFFDL-TR-70-133*. February 1971.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

[704] W. G. Whitman. *Chem. Met. Eng*. 29. 146. 1923.

[705] D. C. Wilcox. *Turbulence Modeling for CFD*. DCW Industries, Inc. La Canada, California. 1998.

[706] D. C. Wilcox. "Formulation of the k–ω turbulence model revisited". *45th AIAA Aerospace Sciences Meeting. Reno, NV, USA. AIAA Paper 2007–1408*. 2007.

[707] J. Wilcox. "On the Path to Elucidating the Speciation of Mercury in the Flue Gases of Coal Combustion". *PhD thesis*. Dept. of Chemical and Environmental Engineering. The University of Arizona, Arizona, U.S.A. January 2004.

[708] J. Wilcox. "A Kinetic Investigation of High-Temperature Mercury Oxidation by Chlorine". *Journal of Physical Chemistry A,*. 113(24). 6633–6639. 2009.

[709] J. Wilcox, D. C. J. Marsden, and P. Blowers. "Evaluation of Basis Sets and Theoretical Methods for Estimating Rate Constants of Mercury Oxidation Reactions Involving Chlorine". *Fuel Processing Technology*. 85(5). 391–400. 2004.

[710] J. Wilcox, J. Robles, D. C. J. Marsden, and P. Blowers. "Theoretically Predicted Rate Constants for Mercury Oxidation by Hydrogen Chloride in Coal Combustion Flue Gases". *Environmental Science and Technology*. 37(18). 4199–4204. September 2003.

[711] F. A. Williams. *Turbulent Mixing in Nonreactive and Reactive Flows*. Plenum Press, New York. 1975.

[712] F. Winter, C. Wartha, G. Loffler, and H. Hofbauer. "The NO and N_2O Formation Mechanism During Devolatilization and Char Combustion Under Fluidized Bed Conditions". *In 26th Symposium (Int'l) on Combustion*. The Combustion Institute. 3325–3334. 1996.

[713] M. Wolfshtein. "The Velocity and Temperature Distribution of One-Dimensional Flow with Turbulence Augmentation and Pressure Gradient". *Int. J. Heat Mass Transfer*. 12. 301–318. 1969.

[714] M. J. Wright, D. Bose, G. E. Palmer, and E. Levin. "Recommended collision integrals for transport property computations, Part 1: Air species.". *AIAA Journal*. 43. 12. 2558–2564. 2005.

[715] H. Wu, X. Li and P. Berg. "On the modeling of water transport in polymer electrolyte membrane fuel cells". *Electrochimica Acta*. 54(27). 6913-6927. 2009.

[716] P.-K. Wu, L.-K. Tseng, and G. M. Faeth. "Primary Breakup in Gas/Liquid Mixing Layers for Turbulent Liquids". *Atomization and Sprays*. 2. 295–317. 1995.

[717] W. Wu, Y. Piao, Q. Xie, and Z. Ren . "Flame Diagnostics with a Conservative Representation of Chemical Explosive Mode Analysis". *AIAA Journal*. 57(4). 295–317. 2019.

[718] V. Yakhot and S. A. Orszag. "Renormalization Group Analysis of Turbulence I Basic Theory". *Journal of Scientific Computing*. 1(1). 1–51. 1986.

[719] W. Yao and C. Morel. "Volumetric Interfacial Area Prediction in Upward Bubbly Two-Phase Flow". *International Journal of Heat and Mass Transfer*. 47. 307-328. 2004.

[720] X. G. Yang, Y. Leng, G. Zhang, S. Ge, and C. Y. Wang . "Modeling of lithium plating induced aging of lithium-ion batteries: Transition from linear to nonlinear aging". *Journal of Power Sources*. 360. 28-40. 2017.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

[721] A. L. Yarin, and D. A. Weiss. "Impact of Drops on Solid Surfaces: Self-similar Capillary Waves, and Splashing as a New Type of Kinematic Discontinuity". *J. Fluid Mech.*. 283. 141-173. 1995.

[722] J. B. Young. "The Spontaneous Condensation of Steam in Supersonic Nozzles". *Physico Chemical Hydrodynamics*. 3(2). 57–82. July 1982.

[723] J. B. Young. "An Equation of State for Steam for Turbomachinery and Other Flow Calculations". *Journal of Engineering for Gas Turbines and Power*. 110. 1–7. January 1988.

[724] J. B. Young. "Two-Dimensional Nonequilibrium, Wet-Steam Calculations for Nozzles and Turbine Cascades". *Journal of Turbomachinery*. 114. 569–579. July 1992.

[725] D. L. Youngs. "Time-Dependent Multi-Material Flow with Large Fluid Distortion". *Numerical Methods for Fluid Dynamics*. K. W. Morton and M. J. Baines, editors. Academic Press. 1982.

[726] X. Zhang, S. Chumakov, X. Li, et al. "An Electro-chemo-thermo-mechanical Coupled Three-dimensional Computational Framework for Lithium-ion Batteries". *Journal of The Electrochemical Society*. 167 (16). 2020.

[727] X. Zhang, M. Klinsmann, S. Chumakov, et al. "A modified electrochemical model to account for mechanical effects due to lithium intercalation and external pressure". *Journal of The Electrochemical Society* . 168 (2). 2021.

[728] Q. Zhou and M. A. Leschziner. "Technical report". *8th Turbulent Shear Flows Symp, Munich*. 1991.

[729] O. C. Zienkiewicz. *The Finite Element Method*. McGraw-Hill Company. London. 1977.

[730] T. Ziegenhein and D. Lucas. "Observations on bubble shapes in bubble columns under different flow conditions". *Exp. Therm. Fluid Sci.*. 85. 248–256. 2017.

[731] T. Ziegenhein, A. Tomiyama, and D. Lucas. "A new measuring concept to determine the lift force for distorted bubbles in low Morton number system: Results for air/water". *International Journal of Multiphase Flow*. 108. 11–24. 2018.

[732] V. Zimont. "Gas Premixed Combustion at High Turbulence. Turbulent Flame Closure Model Combustion Model". *Experimental Thermal and Fluid Science*. 21. 179–186. 2000.

[733] V. Zimont, W. Polifke, M. Bettelini, and W. Weisenstein. "An Efficient Computational Model for Premixed Turbulent Combustion at High Reynolds Numbers Based on a Turbulent Flame Speed Closure". *J. of Gas Turbines Power*. 120. 526–532. 1998.

[734] V. L. Zimont, F. Biagioli, and K. J. Syed. "Modelling Turbulent Premixed Combustion in the Intermediate Steady Propagation Regime". *Progress in Computational Fluid Dynamics*. 1(1). 14–28. 2001.

[735] V. L. Zimont and A. N. Lipatnikov. "A Numerical Model of Premixed Turbulent Combustion of Gases". *Chem. Phys. Report*. 14(7). 993–1025. 1995.

[736] G. Zschaeck, T. Frank, A. D. Burnsb. "CFD modelling and validation of wall condensation in the presence of non-condensable gases". *Nuclear Engineering and Design*. 279. 137–146. 2014.

[737] P. J. Zwart. "Numerical Modelling of Free Surface and Cavitating Flows". *VKI Lecture Series*. 2005.

[738] P. J. Zwart, A. G. Gerber, and T. Belamri. "A Two-Phase Flow Model for Predicting Cavitation Dynamics". *In Fifth International Conference on Multiphase Flow, Yokohama, Japan*. 2004.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

[739] P. J. Zwart, P. G. Godin, J. Penrose, S. H. Rhee. "Ship Hull Simulations with a Coupled Solution Algorithm". In Y. Nagano K. Hanjalic and M. J. Tummers, editors *Proceedings of the 10th International Symposium on Practical Designs of Ships and Other Floating Structures*. Houston, TX. 2007.

[740] A. Arefmanesh, S. G. Advani, and E. E. Michaelides. A Numerical Study of Bubble Growth During Low Pressure Structural Foam Molding Process. *Polym. Eng. Sci.*. 30. 1330–13337. 1990.

[741] H. A. Barnes, J. F. Hutton, and K. Walters. *An Introduction to Rheology*. Elsevier. 1989.

[742] R. B. Bird, R. C. Armstrong, and O. Hassager. *Dynamics of Polymeric Liquids*. John Wiley. 1987.

[743] R. B. Bird, P. J. Dotson, and N. L. Johnson. *J Non-Newtonian Fluid Mech*. 7:213. 1980.

[744] R. B. Bird, W. E. Stewart, and E. N. Lightfoot. *Transport Phenomena*. John Wiley & Sons. 1960.

[745] R. S. Chambers. Numerical Integration of the Hereditary Integrals in a Viscoelastic Model for Glass. *J Am Ceram Soc*. 75(8). 2213–2218. 1992.

[746] N. Clemeur, R. P. G. Rutgers, and B. Debbaut. On the evaluation of some differential formulations for the pom-pom constitutive model. *Rheol Acta*. 42(1). 217–231. 2003.

[747] P. Coussot, A. I. Leonov, and J. M. Piau. Rheology of concentrated dispersed systems in a low molecular weight matrix. *J Non-Newtonian Fluid Mech*. 46. 179–217. 1993.

[748] A. R. Davies. Reentrant corner singularities in non-Newtonian flow. Part I: theory. *J Non-Newtonian Fluid Mech*. 29. 269–293. 1988.

[749] B. Debbaut and M. J. Crochet. Extensional effects in complex flows. *J Non-Newtonian Fluid Mech*. 30. 169–184. 1988.

[750] B. Debbaut and M. J. Crochet. Further results on the flow of a viscoelastic fluid through an abrupt contraction. *J Non-Newtonian Fluid Mech*. 20. 173–185. 1986.

[751] A. K. Doufas, A. J. McHugh, and C. Miller. Simulation of melt spinning including flow-induced crystallization Part I: model development and predictions. *J Non-Newtonian Fluid Mech*. 92. 27–66. 2000.

[752] M. Fortin. Old and New Finite Elements for Incompressible Flows. *Int Numerical Methods Fluids*. 1. 347–364. 1987.

[753] G. S. Fulcher. Analysis of Recent Measurements of the Viscosity of Glasses. *J Am Ceram Soc*. 8(6). 339–355. 1925.

[754] M. Goldstein and J. R. Howell. Boundary Conditions for the Diffusion Solution of Coupled Conduction-Radiation Problems. *Technical Report NASA-TN-D-4618*. NASA Lewis Research Center. 1968.

[755] A. Goublomme, B. Draily, and M. J. Crochet. Numerical Prediction of Extrudate Swell of a High-Density Polyethylene. *J Non-Newtonian Fluid Mech,*. 44. 171–195. 1992.

[756] R. Guénette and M. Fortin. A new mixed finite element method for computing viscoelastic flow. *J Non-Newtonian Fluid Mech*. 60(1). 27–52. 1995.

[757] J. R. Howell and M. Goldstein. Effective Slip Coefficients for Conduction-Radiation Problems. *J Heat Transfer*. 91(1). 165–169. 1969.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

[758] N. J. Inkson, T. C. B. McLeish, O. G. Harlen, and D. J. Groves. Predicting low density polyethylene melt rheology in elongational and shear flows with "pom-pom" constitutive equations. *J Rheol*. 43(4). 873–896. 1999.

[759] C. Johnson. Numerical Solutions of Partial Differential Equations by the Finite Element Method. Cambridge Univ Press. 1987.

[760] R. Keunings. On the high Weissenberg number problem. *J Non-Newtonian Fluid Mech*. 20. 209–226. 1986.

[761] R. Keunings. "Simulation of Viscoelastic Fluid Flow". *Fundamentals of Computer Modeling for Polymer Processing*. 402–470. C. L. Tucker III, editor. Hanser Publishers, Munich. 1989.

[762] A. I. Leonov. Analysis of simple constitutive equations for viscoelastic liquids. *J Non-Newtonian Fluid Mech*. 42. 323–350. 1992.

[763] A. Markovsky and T. F. Soules. An Efficient and Stable Algorithm for Calculating Fictive Temperatures. *Comm Am Ceram Soc*. 67. C-56-C-57. 1984.

[764] T. C. B. McLeish and R. C. Larson. Molecular constitutive equations for a class of branched polymers: The pom-pom polymer. *J Rheol*. 42(1). 82–112. 1998.

[765] O. S. Narayanaswamy. A Model of Structural Relaxation in Glass. *J Am Ceram Soc*. 54(10). 491–498. 1971.

[766] Park, Chul. Review of chemical-kinetic problems of future NASA missions. I-Earth entries. *Journal of Thermophysics and Heat Transfer 7*. 3. 385–398. 1993.

[767] Park, Chul, Howe, Jaffe, Candler. Review of chemical-kinetic problems of future NASA missions. II-Mars entries. *Journal of Thermophysics and Heat Transfer 8*. 1. 9–23. 1994.

[768] D. Rajagopalan, R. C. Armstrong, and R. A. Brown. *J Non-Newtonian Fluid Mech*. 36. 159–192. 1990.

[769] Y. Saad and M. H. Schultz. GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM J Sci Stat Comput*. 7. 856–869. 1986.

[770] R. Siegel and J. R. Howell. *Thermal Radiation Heat Transfer*. McGraw-Hill, New York. 1981.

[771] M. Simhambhatla and A. I. Leonov. On the rheological modeling of filled polymers with particle-matrix interactions. *Rheol Acta*. 34. 329–338. 1995.

[772] M. V. Simhambhatla. The rheological modeling of simple flows of unfilled and filled polymers. *PhD thesis*. University of Akron, Akron, Ohio. 1994.

[773] K. Stueben. Algebraic Multigrid (AMG): An Introduction with Applications. *GMD Report*. 53. 1999.

[774] R. I. Tanner. *Engineering Rheology*. Oxford Universirt Press. 2002.

[775] J. F. Thompson, Z. U. A. Warsi, and C. W. Mastin. Numerical Grid Generation: Foundations and Applications. Elsevier. 1985.

[776] A. Ungan and R. Viskanta. Effects of Air Bubbling on Circulation and Heat Transfer in a Glass Melting Tank. *J Am Ceram Soc*. 69. 382–391.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.

[777] H. A. van der Vorst. Bi-CGSTAB: A fast and smoothly converging variant of Bi-CG for the solution of nonsymmetric linear systems. *SIAM J Sci Stat Comput*. 13. 631–644. 1992.

[778] J. J. Van Schaftingen and M. J. Crochet. *Int J Numerical Methods Fluids*. 4. 1065–1081. 1984.

[779] M. H. Wagner. Constitutive Analysis of Uniaxial Elongational Flow Data of Low-Density Polyethylene Melt. *J Non-Newtonian Fluid Mech*. 4. 39–55. 1978.

[780] Wieghardt, K., & Tillmann, W. (1951): "On the turbulent friction layer for rising pressure". NACA TM 1314..

[781] H-H. Yang and I. Manas-Zloczower. Analysis of mixing. *Int Polymer processing IX*. 1994.

Release 2024 R1 - © ANSYS, Inc. All rights reserved. - Contains proprietary and confidential information of ANSYS, Inc. and its subsidiaries and affiliates.