Tutorial: Sinusoidal Wall Temperature Variation

Introduction

This tutorial examines fluid flow through a two dimensional channel, where one wall of the channel has user-defined temperature profile applied to it. The purpose of this tutorial is to demonstrate the ability of ANSYS FLUENT to use user-defined functions (UDFs) to specify a position-dependent variable on the wall boundary condition.

This tutorial demonstrates how to do the following:

- Interpret the UDF.
- Use UDF for specifying the profiles on boundaries.
- Postprocess the resulting data.

Prerequisites

This tutorial is written with the assumption that you have completed Tutorial 1 from ANSYS FLUENT 12.0 Tutorial Guide, and that you are familiar with the ANSYS FLUENT navigation pane and menu structure. Some steps in the setup and solution procedure will not be shown explicitly.

For more details about UDFs, see ANSYS FLUENT 12.0 UDF Manual.

Problem Description

The schematic of the problem considered in this tutorial is shown in Figure 1.

Air at 300 K enters a 2D channel which has an insulated upper wall and a heated lower wall.
The temperature on the lower wall varies sinusoidally with x-position according to:

$$T(x) = 300 + 100 \sin\left(\frac{\pi x}{0.005}\right)$$

The calculation will be performed assuming steady state, incompressible, and laminar flow in the channel.

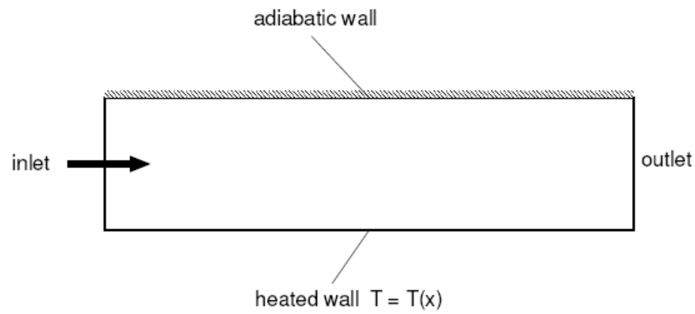


Figure 1: Problem Specification

Setup and Solution

Preparation

1. Copy the files (channel.msh and wallprof.c) to your working folder.
2. Use FLUENT Launcher to start the 2D version of ANSYS FLUENT.

   For more information about FLUENT Launcher *see Section* 1.1.2, Starting ANSYS FLUENT Using FLUENT Launcher *in* ANSYS FLUENT *12.0 User's Guide.*
3. Enable Double-Precision in the Options list.
4. Click the UDF Compiler tab and ensure that the Setup Compilation Environment for UDF is enabled.

   The path to the .bat file which is required to compile the UDF will be displayed as soon as you enable Setup Compilation Environment for UDF*.*

   If the UDF Compiler *tab does not appear in the* FLUENT Launcher *dialog box by default, click the* Show More >> *button to view the additional settings.*

   The Display Options *are enabled by default. Therefore, after you read in the mesh, it will be displayed in the embedded graphics window.*

Step 1: Mesh

1. Read the mesh file (`channel.msh`).

   File ⟶ Read ⟶ Mesh...

   As the mesh file is read, ANSYS FLUENT *will report the progress in the console.*

Step 2: General Settings

1. Retain the default solver settings.

   General

2. Check the mesh (see Figure 2).

   General ⟶ Check

   ANSYS FLUENT *will perform various checks on the mesh and will report the progress in the console. Make sure the minimum volume reported is a positive number.*

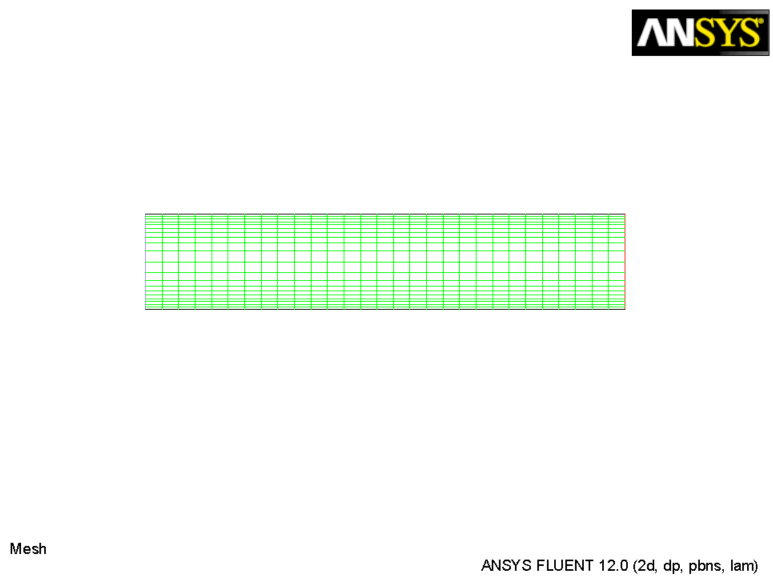


Figure 2: Mesh Display

Step 3: Models

Enable energy equation.

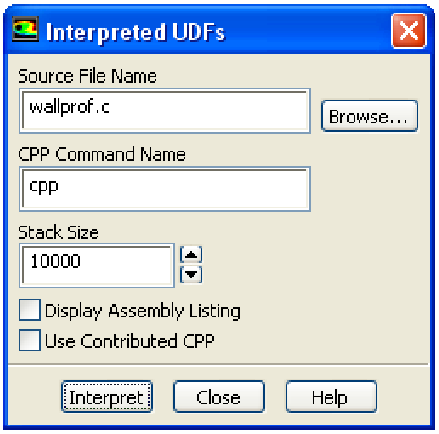
Models ⟶ Energy ⟶ Edit...

You will use the default fluid properties of air for this problem. Therefore, you need not make any changes to the material properties.

Step 4: User-Defined Functions

Interpret the UDF.

Define ⟶ User-Defined ⟶ Functions ⟶ Interpreted...



1. Click on the Browse... button.
2. Select the source file (wallprof.c) in the Select File dialog box.
3. Specify the C preprocessor to be used in the CPP Command Name field.

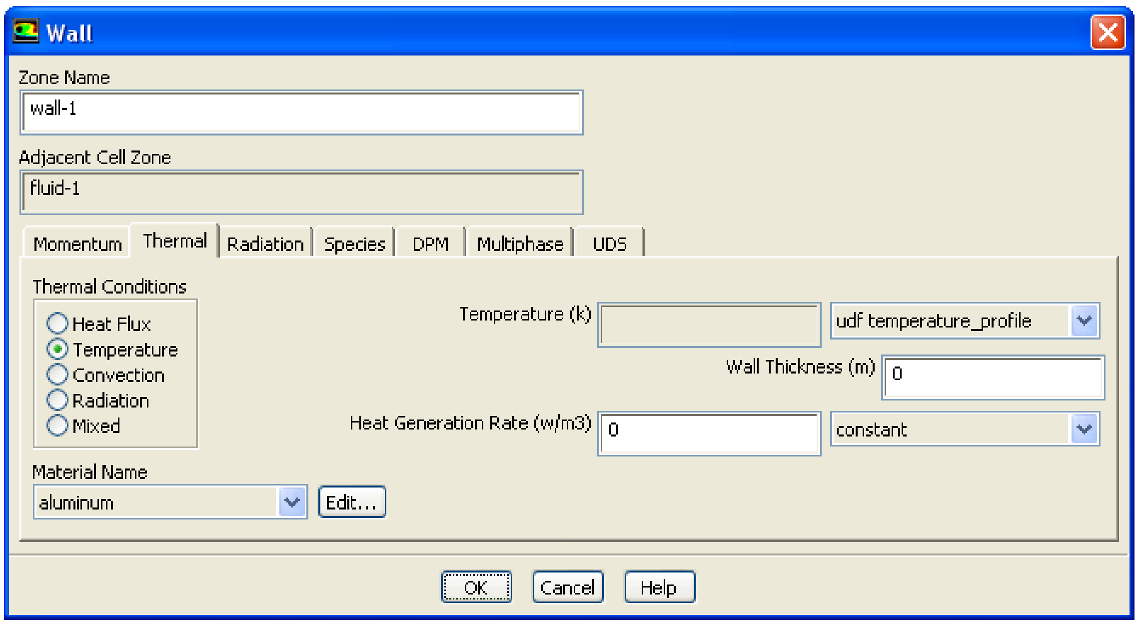
   If you want to use the C preprocessor that ANSYS, Inc. has supplied, you can enable the Use Contributed CPP *option.*
4. Retain the default value of 10000 for Stack Size.

   The Stack Size *should be* 10000 *unless the number of local variables in your function causes the stack to overflow. Its value should be set to a number that is greater than the number of local variables used.*
5. Click Interpret and close the Interpreted UDFs dialog box.

Step 5: Boundary Condition

1. Set the boundary conditions for wall-1.

   Boundary Conditions ⟶ wall-1 ⟶ Edit...

   (a) Click the Thermal tab and select Temperature from the Thermal Conditions list.

   (b) Select udf temperature_profile from the Temperature drop-down list.

   (c) Retain default values for the other parameters.

   (d) Click OK to close the Wall dialog box.

2. Set the boundary conditions for the velocity-inlet-1 zone.

   Boundary Conditions ⟶ velocity-inlet-1 ⟶ Edit...

   (a) Select Components from the Velocity Specification Method drop-down list.

   (b) Enter 1 m/s for X-Velocity.

   (c) Click OK to close the Velocity Inlet dialog box.

Step 6: Solution

1. Change the Absolute Criteria for continuity to 0.0001.

   Monitors ⟶ Residuals ⟶ Edit...

2. Initialize the flow field from velocity-inlet-1.

   Solution Initialization

3. Save case and data files (`channel.cas/dat.gz`).

4. Start the calculation for 100 iterations (Figure 3).

   Run Calculation

   The solution converges in approximately 60 iterations.

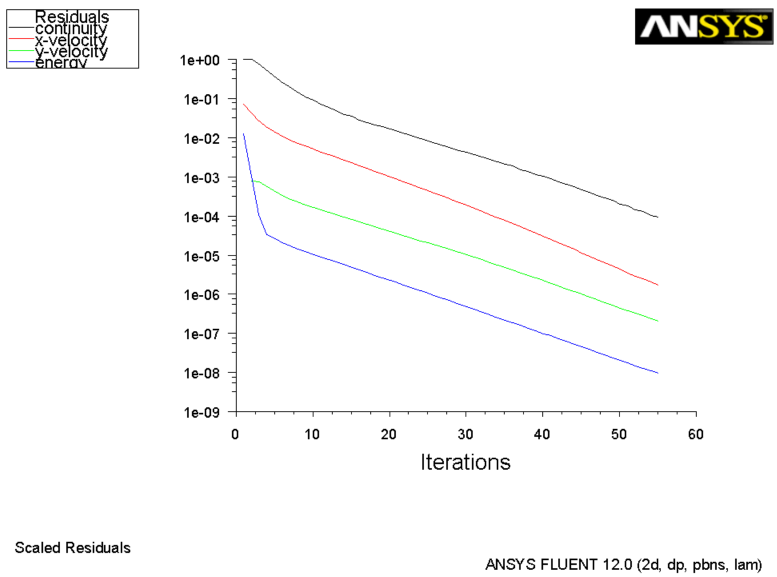


Figure 3: Scaled Residuals

Step 7: Postprocessing

Display filled contours of static temperature.

Graphics and Animations ⟶ Contours ⟶ Set Up...

1. Enable Filled in the Options list.

2. Select Temperature... and Static Temperature from Contours of drop-down list.

3. Click Display (see Figure 4).

4. Close Contours dialog box.

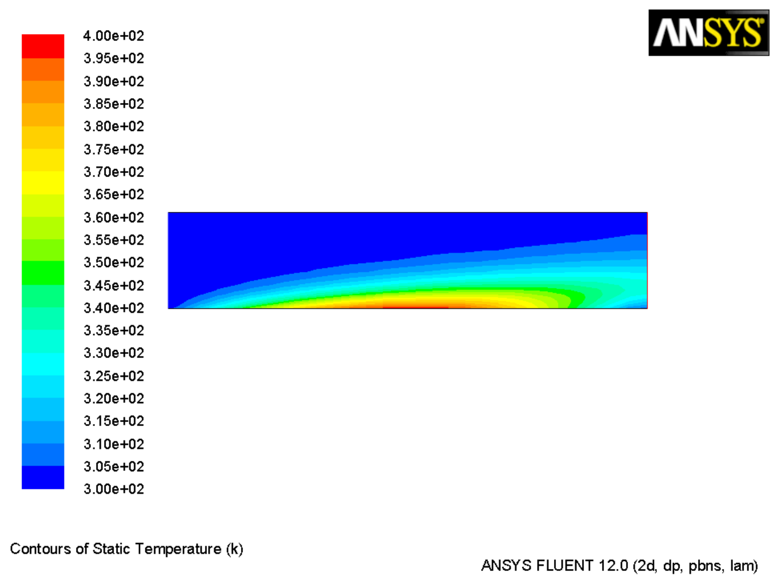


Figure 4: Contours of Static Temperature

Appendix

The UDF (wallprof.c) is used to specify a sinusoidal temperature variation on the wall. In a separate editor, you can view the contents of the UDF to understand its structure and function.

The contents of the UDF file are as follows:

```
/*****************************************************************/
/*                                                               */
/* User-Defined Functions for sinusoidal temperature profile    */
/*                                                               */
/*****************************************************************/

#include "udf.h"
#define PI 3.141592654

DEFINE_PROFILE(temperature_profile, thread, position)
 real r[3]; /* this will hold the position vector */
 real x;
 face_t f;

   begin_f_loop(f, thread)

     F_CENTROID(r,f,thread);
     x = r[0];
     F_PROFILE(f, thread, position) = 300.+100.*sin(PI*x/0.005);

   end_f_loop(f, thread)
```


Results

The contour plot in Figure 4 shows that the temperature on the wall and in the fluid reaches a peak at the center of the channel due to the peak in the prescribed wall temperature.

Summary

The tutorial demonstrated the use of UDFs for specifying profiles on boundaries. You can apply this approach to the other boundary condition types such as pressure and velocity inlets, and pressure outlets.
When you are comfortable with this problem, try modifying the UDF to specify heat flux rather than temperature.

- Replace the following line in the udf code with the specified heat flux expression.

$$temp = 300. + 100.*sin(pi * x/0.005)$$

- Change the boundary condition type.
- Hook the new profile to the wall boundary before running the new case.