
Tutorial: Position Dependent Porous Media

Introduction

The purpose of this tutorial is to illustrate the modeling of a position dependent porous plug in a two dimensional channel flow with the help of a user-defined function (UDF).

This tutorial demonstrates how to do the following:

- Compile the UDF.
- Use UDF for specifying the source terms.
- 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 problem considered in this tutorial is shown schematically in Figure 1.

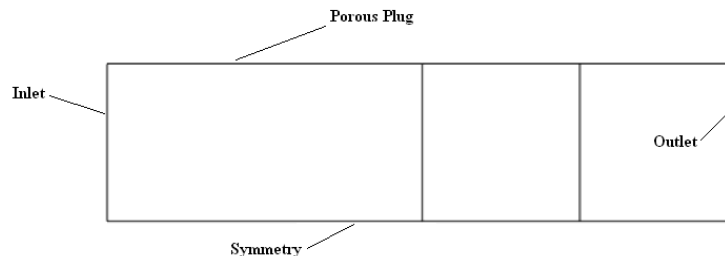


Figure 1: Schematic of the Problem

The domain is divided into two fluid zones. In the fluid-2 cell zone, an X-momentum source term is included for each cell using UDF. This source term models the effect of a porous plug on the flow in the X-direction and is given by:

$$S_x = -\frac{1}{2}C\rho y|u|u \quad (1)$$

where $C = 100$ is a model constant.

Equation source terms are added using the `DEFINE_SOURCE` macro.

The finite-volume solver of ANSYS FLUENT expects the source term to be *linearized* according to the following convention:

$$S_\phi = A + B\phi = \underbrace{\left(S^* - \left(\frac{\partial S_\phi}{\partial \phi}\right)^* \phi^*\right)}_A + \underbrace{\left(\frac{\partial S_\phi}{\partial \phi}\right)^*}_{B} \phi \quad (2)$$

where the superscript $*$ represents the value at the previous iteration. B (called `dS[eqn]` in the UDF macro) can be coded explicitly by using currently known value of ϕ , and the entire S is returned by `DEFINE_SOURCE`. For the simple source of X-momentum equation in this problem,

$$B = \frac{\partial S_x}{\partial u} = -C\rho y|u|. \quad (3)$$

For more details, examine the source code (`porous_plug.c`) as shown in the [Appendix](#).

Setup and Solution

Preparation

1. Copy the files (`porous_plug.msh` and `porous_plug.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 (porous_plug.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.



Mesh

ANSYS FLUENT 12.0 (2d, dp, pbns, ske)

Figure 2: Mesh Display

Step 3: Models

Select the k-epsilon turbulence model.

Models → Viscous → Edit...

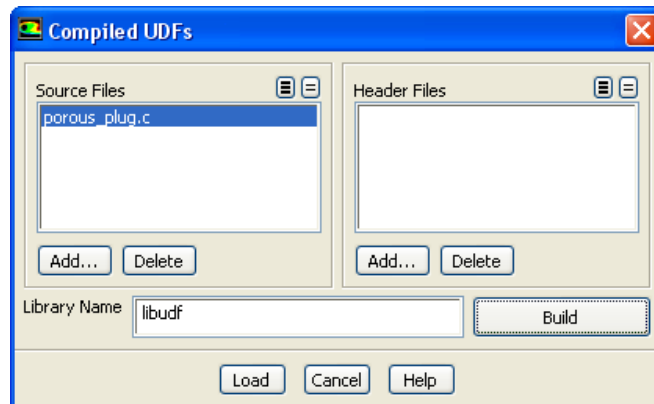
You will use the default fluid properties of air for this problem. Hence, you need not make any changes to the material properties.

Step 4: User-Defined Functions

The UDF can be compiled as well as interpreted. In this tutorial, use the compiled option.

Compile the UDF (porous_plug.c).

Define → User-Defined → Functions → Compiled...



1. Click Add... and select the source file, porous_plug.c.

2. Click Build to build the library.

A Warning dialog box opens, asking you to ensure that the UDF source files are in the same folder that contains the case and data files. Click OK.

3. Click Load to load the newly created UDF library.

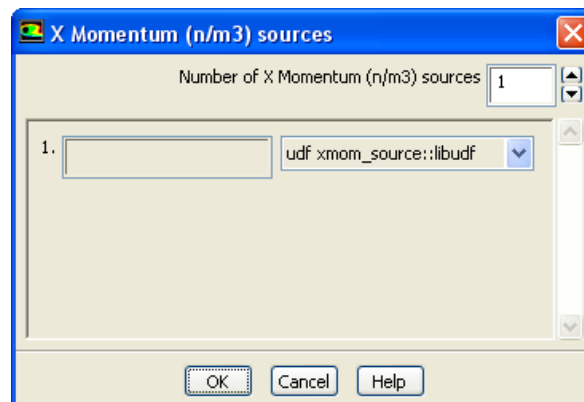
Step 5: Cell Zone Conditions

Set the cell zone conditions for the fluid-2.

Cell Zone Conditions → fluid-2 → Edit...

1. Click the Source Terms tab and enable Source Terms.

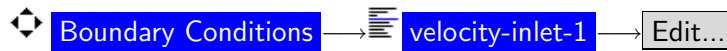
2. Click Edit... for X Momentum (n/m³) to open the X Momentum (n/m³) sources dialog box.



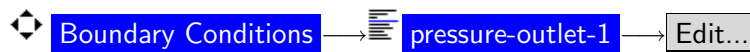
- (a) Set the Number of X Momentum (n/m3) sources to 1.
 - (b) Select `udf xmom_source::libudf` from the drop-down list.
 - (c) Click OK to close the X Momentum (n/m3) sources dialog box.
3. Click OK to close the Fluid dialog box.

Step 6: Boundary Conditions

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



- (a) Enter 1 m/s for Velocity Magnitude.
 - (b) Select Intensity and Hydraulic Diameter from the Specification Method drop-down list in the Turbulence group box.
 - (c) Enter 5% for Turbulent Intensity and 4 m for Hydraulic Diameter.
 - (d) Retain the default values for the other parameters.
 - (e) Click OK to close the Velocity Inlet dialog box.
2. Set the boundary conditions for pressure-outlet-1.



- (a) Retain 0 pascal for Gauge Pressure.
 - (b) Select Intensity and Viscosity Ratio from the Specification Method drop-down list in the Turbulence group box.
 - (c) Enter 5% for Backflow Turbulent Intensity and 10 for Backflow Turbulent Viscosity Ratio.

- (d) Retain the default values for other parameters.
- (e) Click OK to close the Pressure Outlet dialog box.

Step 7: Solution

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



2. Start the calculation for 100 iterations (see Figure 3).



The solution converges in approximately 30 iterations.

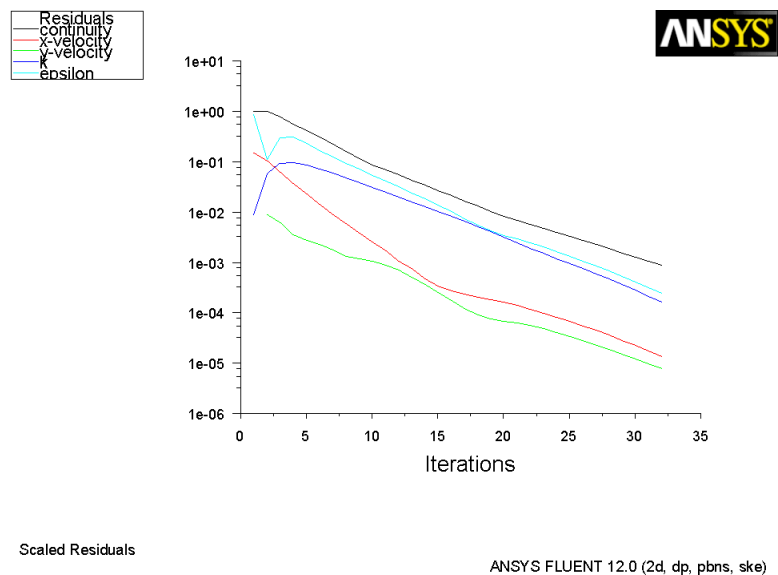
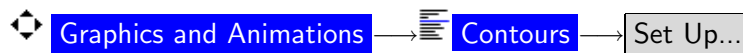


Figure 3: Scaled Residuals

Step 8: Postprocessing

1. Display the pressure contours.



- (a) Enable Filled in the Options list.
- (b) Select Pressure... and Static Pressure from Contours of drop-down list.
- (c) Click Display (Figure 4) and close the Contours dialog box.

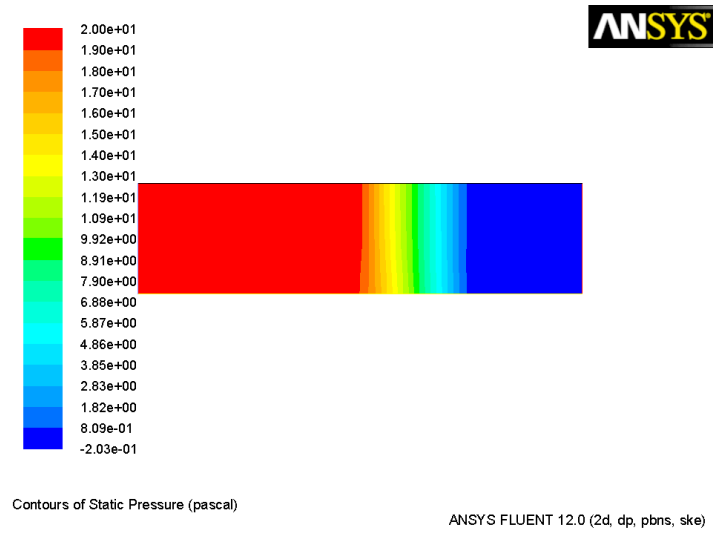


Figure 4: Pressure Contours

2. Display the velocity vectors (see Figure 5).

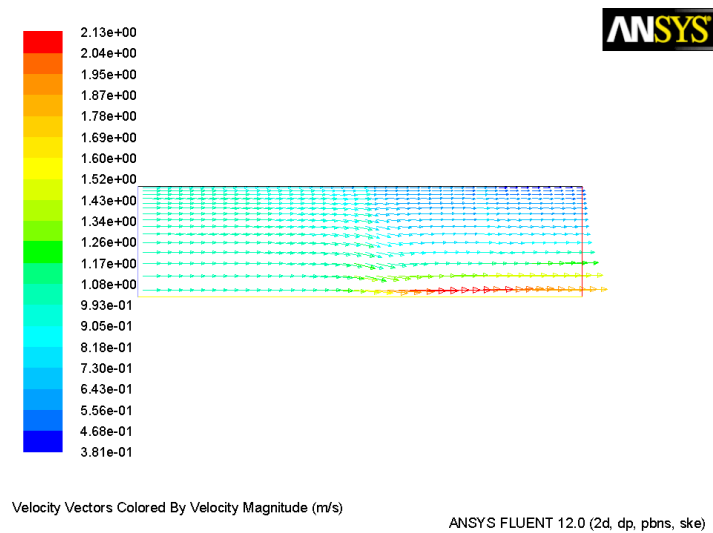
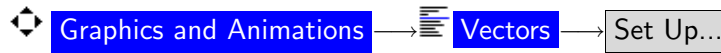


Figure 5: Velocity Vectors Colored by Velocity Magnitude

Appendix

The contents of the UDF source code are as follows:

```
/* **** */
/* Position-Dependent Porous Media */
/* **** */

#include "udf.h"

DEFINE_SOURCE(xmom_source, cell, thread, dS, eqn)
{
    const real c2=100.0;
    real x[ND_ND];
    real con, source;

    C_CENTROID(x, cell, thread);
    con = c2*0.5*C_R(cell, thread)*x[1];
    source = - con*fabs(C_U(cell, thread))*C_U(cell, thread);
    dS[eqn] = - 2.*con*fabs(C_U(cell, thread));

    return source;
}
```

Results

Figure 4 shows that fluid-2 is causing pressure drop in the system and thus acting as a porous plug. Figure 5 shows that the fluid flows preferentially near the bottom of the channel due to the effect of the source term on the axial momentum of the fluid.

Summary

This tutorial demonstrated the use of UDF for specifying source terms. User-defined source terms can be useful for introducing additional physics into the simulation, and can usually be modeled as sources or sinks of mass, momentum, energy, species etc.

Extra: *When you are comfortable with the present exercise, you can try modifying the UDF to specify momentum sources for both the X and Y momentum equations. The present UDF can be copied and suitably modified for the Y momentum equation. Both functions will then reside in the UDF file, and be accessible to the solver upon compilation.*