Fluent – Adding the Energy Equation to 2-D Couette flow

Author: John M. Cimbala, Penn State University
Latest revision: 24 September 2007

Introduction and Instructions:
In this document is a procedure that enables you to add and solve the energy equation in a simple Couette flow problem with the CFD program, Fluent. This set of instructions assumes that you have already run the two prerequisite learning modules, in which a grid was generated and a flow solution was found for the steady, incompressible, 2-D Couette flow problem.

Log on and launch Fluent:
1. Log on to any computer with a valid Fluent license.
2. Start Fluent, and select the 2ddp option (two-dimensional, double precision). Run.

Read the case and data files from the Couette flow problem:
1. Select File-Read-Case & Data. Browse as necessary to find the case file you had saved previously. Highlight the case file you saved previously for 2-D Couette flow, and OK. Fluent will read in the grid geometry and mesh that were previously created by Gambit, and it will also read the calculations of velocity and pressure that were stored in the previous run of Fluent. Some information is displayed on the main screen. If all went well, it should give no errors, and the word Done should appear.
2. Look at the velocity profile to make sure it is correct. In the main Fluent window, Plot-XY Plot. In the new window that pops up, *Solution XY Plot*, make sure the following options are turned on: Node Values, Position on Y Axis. (Position on X Axis must be turned off.) Set the Plot Direction to 0 for *X* and 1 for *Y*, which ensures that the desired flow variable will be plotted versus *y*-location.
3. In X Axis Function, select Velocity, and in the drop-down box below that, change Velocity Magnitude to X-Velocity. This sets up the plot for velocity component *u* versus position *y*, which is desired.
4. In the *Surfaces* box at the lower righ, highlight vertical (or whatever you called the vertical line previously defined in the center of the channel). This should be the *only* surface highlighted.
5. Click on Plot to show the profile. Verify that this is the same linear velocity profile that was calculated previously. If it is not, you may have to repeat the calculations.
6. Move the *Solution XY Plot* to the side, since it will be needed later.

Turn on the energy equation and define the Case 1 thermal boundary conditions:
1. In the main *Fluent* window, Define-Models-Energy.
2. In the *Energy* window that pops up, turn on Energy Equation, and OK. This will turn on the energy equation.
3. In the main *Fluent* window, Define-Models-Viscous. Turn on Viscous Heating. OK.
4. Now the thermal boundary conditions need to be specified. In the main *Fluent* window, click on Define-Boundary Conditions, and the *Boundary Conditions* window will pop up.
5. Select the edge corresponding to the bottom wall of the channel, and Set. There are additional thermal boundary condition choices which were not there when this problem was solved previously without heat transfer. In the Thermal tab, select Temperature, which means a constant temperature boundary condition, and 300 K, which should be the default wall temperature. The wall material can be any solid material - the default (aluminum) is fine, so OK.
6. Select the upper wall, and Set. This is the top wall of the channel, which moves at 1.0 m/s. The moving wall option should already be *on*, with a wall speed specified as 1 m/s. Set the thermal boundary condition to be a fixed temperature of 300 K, just as was done for the lower wall, and OK.
7. Now Close the Boundary Conditions window.

Check the residual parameters and run the constant temperature case:
1. In the main *Fluent* window, Solve-Monitors-Residual. In the *Residual Monitors* window, both Plot and Print should already be turned on in the *Options* portion of the window. Notice a fourth residual for the energy equation. Turn off (un-check) the Check Convergence options for *all* the equations. This will enable you to iterate to machine accuracy. OK.
2. To open up the *Iterate* window, Solve-Iterate. Change *Number of Iterations* to 200, change the *Reporting Interval* to 10, and Iterate. The main screen will list the residuals after every 10 iterations, while the graphics display window will plot the residuals as a function of iteration number.
3. Keep iterating, monitoring the residuals. After a while the residuals should all level off at very low values. You may have to iterate for more than a thousand iterations, but it should not take very long.

Generate a plot of the temperature profile, and save the profile data to a text file:

1. Return to the *Solution XY Plot* window. If you can't find it under all your other windows, go to the main *Fluent* window, and Plot-XY Plot. Change the *X Axis Function* to Temperature..., and under that, Static Temperature.
2. In the *Surfaces* box at the lower right, highlight vertical (or whatever you called the vertical line previously defined in the center of the channel). This should be the *only* surface highlighted. Plot the profile.
3. Unfortunately, since the temperature differences are only a fraction of a degree, the resolution of the temperature axis is not very good. Specifically, there may not be enough variation in temperature for the *x*-axis to be labeled properly. Select Axes, and change *Type* to Float, change *Precision* to about 6 (to get more significant digits on the plot axes). Apply, Close, and then Plot on the *Solution XY Plot* window.
4. If this still does not work, enlarge the graphics window to make the *x*-axis longer. You may also turn off the Auto Range option in the *Axes* window and fill in your own minimum and maximum temperatures.
5. Actually, it makes more sense to plot temperature *difference* than the magnitude of temperature itself. Here is how to create a customized function for the temperature difference:
    1. In the main *Fluent* window, Define-Custom Field Functions. A new window, called *Custom Field Function Calculator* opens up.
    2. In the *Field Functions* area, Temperature, and below that, Static Temperature. Select.
    3. Now use the calculator pad to subtract 300 (click on the virtual keys -, 3, 0, and 0). At this point the display on the top of the calculator pad should say "temperature - 300".
    4. Name this customized function ("temperature-difference" is a recommended name), and Define.
    5. Finally, Close the *Custom Field Function Calculator* window.
6. Return to the *Solution XY Plot* window. In the *X Axis Function* area, choose Custom Field Functions. The new function you just created will appear right below that. (If you create more than one custom field function, all of them will appear there as an option.)
7. Look at the temperature difference profile data by clicking on Plot.
8. These temperature difference data should also be saved as a text file so that later on, a plot comparing the calculations to theory or these data to those produced by different boundary conditions can be generated. In the *Options* section of the *Solution XY Plot* window, turn on Write to File and Write.
9. Make up a file name when prompted (it should end in .txt), and OK.

Save your calculations:

1. In the main *Fluent* window, File-Write-Case & Data. Save as binary and compressed (filename.cas.gz) to save disk space.
2. Fluent will save two files: case (the grid plus all boundary conditions and other specified parameters) and data (the velocity and pressure fields calculated by the code).

Case 2: Re-calculate for the insulated wall boundary condition:

1. Change the thermal boundary condition on the lower wall to Heat Flux. The default value of Heat Flux is 0 (perfect insulation), which is desired, so OK.
2. Iterate until the temperature profile converges to a nice smooth profile for this case.
3. Save the temperature difference profile data for this insulated lower wall case in some new text file (use a different file name so as not to overwrite the previous data).

Save your calculations and exit Fluent:

1. Save these calculations (case and data) as a different file name.
2. Exit Fluent: In the main *Fluent* window, File-Exit.