Fluent – Incompressible 2-D Couette flow

Author: John M. Cimbala, Penn State University
Latest revision: 19 September 2007

Introduction and Instructions

In this document is a procedure that enables you to solve a simple two-dimensional Couette flow problem with the CFD program, Fluent. *Note*: This set of instructions assumes that you have already run Gambit (Fluent's grid generation program), and have generated a grid for channel flow/Couette flow. The file channel.msh, couette.msh, or some other name (whatever you named it) is assumed to exist in a folder or in the Temp directory.

Log on and launch Fluent:

1. It is assumed that you just completed the Gambit module, and are still logged in.
2. Start Fluent, and select the 2ddp option (two-dimensional, double precision). Run.

Read the grid points and geometry of the channel flow domain:

1. Select File-Read-Case. There should be a file shown on the right side called couette.msh (or whatever else you named your mesh file). If not, you may need to browse to the appropriate folder (directory) where the mesh file had been saved.
2. Highlight the mesh file (click on it), and OK. Fluent will read in the grid geometry and mesh that was previously created by Gambit. Some information is displayed on the main screen. If all went well, it should give no errors, and the word *Done* should appear. If you receive a "fatal signal error", something is obviously wrong with your grid. In some cases, lack of disk space can cause an error like this. Delete some files and try again. In the worst case, re-generate the grid and try again.
3. Check the validity of the grid by Grid-Check. Errors like negative volumes, etc. will be found here if there is a problem with the grid. Notice also the minimums and maximums, which are in millimeters.
4. At this point the grid dimensions must be converted to meters (recall, the grid was generated in millimeters). Grid-Scale. In the *Units Conversion* section of the *Scale Grid* window that pops up, select mm as "Grid Was Created In". The scale factors will be automatically set to 0.001, which is the proper factor to convert from mm to m.
5. Use Scale to actually scale the grid. Close the Scale Grid window.
6. Look at the grid to make sure it is correct. Click Display-Grid-Display. A new window called *Grid Display* opens up showing the grid for the channel flow. If this window is too big, rescale it by dragging the edges of the window. It is best if the graphical display window is small enough that both it and the Fluent window are visible simultaneously. The Fluent window and/or the graphical display window may need to be moved to accomplish this.
7. The graphical display can be zoomed-in or zoomed-out with the *middle* mouse button. If you start on the *upper left* and draw a rectangle with the middle mouse button towards the lower right, the display will zoom in on what is included in the rectangle. Meanwhile, the *left* mouse button can be used to drag the image to a new location. If you draw a rectangle *backwards* with the middle mouse button (from the lower right to the upper left), it will zoom out. Play with this feature until your domain is shown nicely in the window. Close the *Grid Display* window; the graphical display window itself will remain.

Define the boundary conditions:

1. Next, the boundary conditions need to be specified. In Gambit, the boundary conditions were declared: wall, periodic, etc., but actual values for wall velocities, inlet velocities, etc. were never defined. This must be done in Fluent.
2. In the main Fluent window, click on Define-Boundary Conditions, and a new *Boundary Conditions* window will pop up.
3. Select the bottom wall of the channel. This wall should be defined as a stationary wall by default. This can be verified with the Set button, then make sure the moving wall option is *not* checked. Nothing needs changed, so OK.
4. Select the top wall of the channel, which needs to move at 1.0 m/s. Set. Turn the moving wall option *on*, and change the wall speed from 0 to 1. Everything else is fine by default, so OK.
5. Verify that the left and right edges are periodic, and the default is translational periodicity.
6. The fluid needs to be defined. In this problem, let's use liquid water at room temperature. Select fluid, and Set. The default fluid is air, and air is the only option available at this point. However, we can add water-liquid as an option by the following procedure:

7. In the main Fluent window, Define-Materials-Fluent Database. Scroll in the list of materials to water-liquid, and highlight it. The properties of this fluid are displayed. Copy and Close this window. This adds water-liquid to the list of available fluids in the Materials window.
8. Return to the *Materials* window, where water-liquid can now be selected under “Fluent Fluid Materials”. Write down the properties (density and viscosity, along with their units, for future reference.) Close.
9. At this point, the fluid has been added to the list of available materials, but unfortunately, the fluid type for this problem has still not yet been set! Return to Define-Boundary Conditions-fluid-Set. (*Note*: This window may actually still be open, so if you can find it, just go to it. If you can't find it because it is hidden under all your other windows, just re-open it by clicking on Define-Boundary Conditions-fluid-Set.) Select water-liquid as the “Material Name” in the *Fluid* window. OK.
10. Since everything is fine now, Close the Boundary Conditions window. Finally, the fluid has been defined as liquid water.

Set up some parameters and initialize:

1. Fluent does not allow you to start iterating until you initialize. The solution scheme in the CFD code starts with some wrong solution, and iterates towards a more correct solution. It needs an initial guess to start with.
2. In the main Fluent window, Solve-Initialize-Initialize. The default initial values of velocity and gage pressure are all zero, which are fine for the problem at hand. Init and Close. This sets the velocity to zero everywhere in the flowfield (except the upper wall, of course, since it is moving to the right). As the code iterates towards a solution, the effect of the moving upper wall will be felt deeper and deeper into the fluid.
3. Now the convergence criteria need to be set. As the code iterates, residuals are calculated for each flow equation. These residuals represent a kind of average error in the solution – the smaller the residual, the more converged the solution. In the main window, Solve-Monitors-Residual. In the *Residual Monitors* window that pops up, turn on Plot in the Options portion of the window. The Print option should already be on by default. Here, Print refers to text printed in the main Fluent window, and Plot causes the code to plot the residuals on the screen while the code is iterating.
4. Since there are three differential equations to be solved in a two-D incompressible problem, there are three residuals to be monitored for convergence, one each for the continuity, *x*-momentum, and *y*-momentum equations. Fluent lists these as continuity, x-velocity, and y-velocity. The default convergence criteria are 0.001 for all three of these. Experience has shown that this value is generally not low enough for proper convergence. Click on each 0.001 individually, and change it to 0.000001. When all three are set, OK to exit the *Residual Monitors* window. Finally, the code is ready to iterate!

Save your work so far:

1. Every so often, you should save your work so that you can recover in case of a computer problem. In Fluent, a CFD solution is saved in two separate files: a “case” file, which contains information about the grid, boundary conditions, additional surfaces or lines generated, other specified parameters, etc., and a “data” file, which contains the actual values of velocity, pressure, etc. Once the case and data files are saved, they can be re-opened at a later time.
2. At this time, the data (solutions) are the initialized data. In the main Fluent window, File-Write-Case & Data. In the *Select File* window that pops up, the default file name should be couette.cas – you can rename it and browse to a different folder (directory) if so desired.
3. To save disk space, make sure the option Write Binary Files is turned on. To *really* save disk space, add a “.gz” extension to the file name (e.g., name the file “couette.cas.gz”), which writes a binary *compressed* file.
4. Click OK to write the case and data files.
5. Later on, after you have performed some iterations, it is a good idea to save your work again. When a file already exists, Fluent asks you if you want to overwrite it. It is normally okay to overwrite a previous file.

Iterate towards a solution:

1. Use Solve-Iterate to open up the Iterate window. Change Number of Iterations to 10, and Iterate. The main screen will list the residuals after every iteration, while the graphics display window will plot the residuals as a function of iteration number. The residuals will slowly fall.
2. At the end of 10 iterations, check to see how the solution is progressing. In the main Fluent window, Display-Vectors-Display. The graphical display window will show the velocity vectors. Zoom in with the middle mouse, as described above, to view the velocity field in more detail if desired. Has the solution converged enough to agree with the known exact analytical solution?

3. If necessary for a nicer vector plot, adjust the vector *Options* in the *Vectors* window. Scale up or down to increase or decrease the size of the arrows. In the *Vectors* window, Display must be clicked again to update the plot each time the scale factor is changed.
4. Iterate some more - (To restart the iteration, either find the *Iterate* window, which is probably hidden under some other windows at this point, or click again on Solve-Iterate to re-open the *Iterate* window.) In the Iterate window, Change Number of Iterations to 200, change Reporting Interval to 10 since there is no need to display every iteration, and Iterate.
5. Check the velocity vectors, as described above after every 200 iterations. It is wise to move the Iterate window someplace out of the way of the other windows so you can easily restart the iteration. The residuals may go up sometimes. This is normal, as the code attempts to zero in on a solution - after a while the residuals should either level off or reach the convergence criteria we had set.

Create a vertical line at the center of the channel:

1. Fluent enables a user to define a surface (in 2-D a line actually) at any desired location. In this example, a vertical line (parallel to the *y*-axis) will be defined at the center of the channel. This line will be useful for creating a velocity profile, which can be plotted. The profile data can also be saved along this line for comparison and post-processing.
2. Use Surface-Line/Rake to open up a window that will enable a line to be generated in the flow field.
3. Create a vertical line at the center of the channel by entering its end points, from $(x,y) = (0.01,0)$ m to $(0.01,0.001)$ m. Note that the dimensions here are in meters since we have converted from mm to m. Label this line something descriptive, like “vertical” or “vert_center”.
4. Finally, Create to actually create the line specified by its end points. Close.

Generate and save the velocity profile data at the channel vertical centerline:

1. In this section, a velocity profile at the newly formed line will be generated. These data will be plotted, and will also be saved on a text file for post-processing.
2. In the main Fluent window, Plot-XY Plot.
3. 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*.)
4. 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.
5. In the text box X Axis Function, select Velocity, and in the text box below this, change Velocity Magnitude to X-Velocity. This sets up the plot for velocity component *u* versus position *y*, which is desired.
6. Finally, highlight the vertical surface just created in the *Surfaces* box at the lower right to select. Click on Plot to show the profile. Is it linear?
7. Next, these data are to be written to a text file for post-processing. Turn on Write to File in the *Solution X-Y Plot* window.
8. Click Write, and name the file something descriptive like “vertical.txt”. OK.

Generate streamlines and save streamline profile values for post-processing:

1. Fluent has an option for displaying streamlines. In this simple Couette flow example, the streamlines should all be flat horizontal lines; in other more complicated flows, the streamline pattern can be very useful when analyzing results. In this section you will learn how to display streamlines graphically, and how to generate a profile plot of stream function.
2. In the main Fluent window, Display-Contours.
3. Choose *Contours of* Velocity.., and below that choose Stream Function. The default values for other controls are fine as is.
4. To see the range of values of the stream function and its units, Compute. Notice the strange units. Fluent defines a compressible form of the stream function, specifically $\rho u = \partial\psi / \partial y$, so the units shown are kg/s.

   Even these seem not to be correct because this is a 2-D problem; the actual units are kg/(m·s), i.e., kg/s per unit depth into the screen. However, Fluent treats 2-D problems as 3-D problems with unit depth in the *z*-direction; the units of kg/s are valid with this three-dimensionality in mind.
5. To see a plot of the streamlines, Display. Note that lines of constant stream function *are* streamlines by definition, so a contour plot of stream function shows the streamlines.
6. Are the streamlines straight and horizontal as expected? Are the streamlines equally spaced? Why or why not? [Please discuss this as part of your homework assignment.]
7. Generate an XY plot of stream function verses *y*-position at the vertical centerline, and save the data points to a text file for later post-processing. To do this, follow the instructions above for generating the velocity

profile plot and the velocity profile data file. The only difference here is to change the *X Axis Function* to Velocity, and below that choose Stream Function. Note that if "Write to File" is turned on in the Options section in the *Solution XY Plot* window, the "Plot" button at the bottom of that window turns into a "Write" button. Toggle this option on and off to either write data to a file or plot data to the screen. Save the stream function profile data to a file ("stream_function.txt" or some other descriptive file name is suggested).

Calculate forces on the lower and upper walls:

1. Fluent has an option for calculating the force on any wall. In this example, the forces on the bottom and top walls of the channel will be calculated and compared.
2. In the main Fluent window, Report-Forces. Select (highlight with the mouse) only the bottom wall of the channel. Since we are interested only in the component of force in the *x*-direction, the default *Force Vector* of (1,0,0) is appropriate.
3. To calculate and view the force, Print. The forces (both pressure and viscous components) are displayed on the main Fluent window. Record the total force on the bottom wall. (*Note*: The display gives units of N rather than N/m, since this is a 2-D problem but Fluent assumes unit depth in the *z* direction.)
4. Repeat for the top wall. Is the force on the top wall the same as that on the lower wall? Why or why not?

Save your calculations and exit Fluent:

1. In the main Fluent window, File-Write-Case & Data, and save your case and data files.
2. Click OK to overwrite the files onto your directory.
3. Exit Fluent by File-Exit.