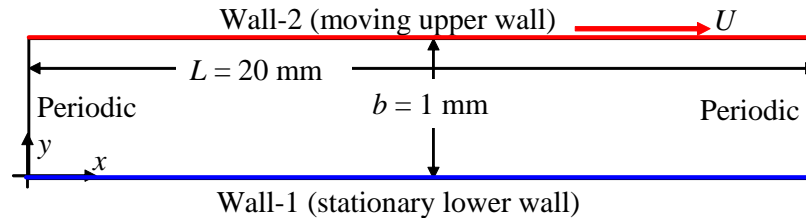


Gambit – Generation of a 2-D channel flow grid

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

Introduction and Instructions

- This document provides a procedure to generate a simple two-dimensional structured rectangular grid, which is used to solve Couette and channel flow problems using the CFD program Fluent.
- The channel is 1 mm high and 20 mm long, as shown in the sketch.



- *Note:* We generate the geometry in millimeters here for convenience, but then later we convert to meters in Fluent.

Log on and start Gambit

1. Log onto any Windows or Linux computer that has a valid license for the Ansys-Fluent family of programs – Gambit (grid generator) and Fluent (CFD solver).
2. Start Gambit. *Note:* If Gambit does not open properly, you may not have the proper path set up in your user files. If this is the case, check with the system administrator.
3. In Gambit, when the mouse cursor is placed over a button, a short description of the function of that button appears in the lower right window under *Description*. This is useful in implementing the commands given below.

Create some data points (vertices) for the desired geometry

1. Create a vertex at the origin, which is the center point of the slot. In *Geometry*, Vertex Command Button-Create Vertex-From Coordinates. In *Create Real Vertex-Global*, enter 0, 0, and 0 for x, y, and z coordinates respectively. [*Optional: Type an appropriate label for this vertex, e.g., “origin”.*] Apply. A vertex is created, as indicated by a small plus sign at this location.
2. Create another vertex with coordinates (20,0,0). [*Optional: Name the vertex if you want.*] Apply.
3. At this point, it is wise to zoom in somewhat to see the vertices more clearly. There are two ways to do this. You can zoom in and move with the right mouse and middle mouse, respectively. Or, to fit all the geometry into the available screen space (a very handy tool!), in *Global Control* (near the bottom right), Fit to Window. After you zoom out, another small plus sign should appear on the screen to the right of the first one.
4. In similar fashion, enter the coordinates of two other points defining the limits of the computational domain, (20,1,0) and (0,1,0).
5. Click on the Fit to Window button again since we are done entering vertices, and have defined the extent of the computational domain.
6. In the main Gambit window near the upper left, File-Save As. Browse to a directory (folder) of your choice, and type in a file name. This saves your work so far. It is a good idea to do this every so often, especially after a major task is completed.

Create edges from these vertices to define the computational domain

1. Next, some edges are created to define the computational domain. Under *Geometry*, Edge Command Button.
2. Under *Edge*, R-Create Edge-Straight. A new window called *Create Straight Edge* appears. (*Note:* If the button is already open, clicking it again will make the window disappear! If this happens, click it a second time to restore it.)
3. In *Create Straight Edge*, make sure the text area called *Vertices* is highlighted, and the cursor is positioned there. If not, click in that area. **Vertices are selected by Shift + left mouse click**. Select the bottom left and then the top left vertices *in order*.
4. Type in an appropriate label for the edge about to be created. I suggest “left” or something equally descriptive. Apply. Gambit will create a straight edge between these two vertices. A yellow line should be drawn on the screen connecting the vertices.

5. In similar fashion, create the right edge, being sure to select the bottom right vertex first, followed by the top right vertex (the order is important).
6. Create the other two edges: bottom and top. Be sure to select the left vertex and then the right vertex (in order) so that the arrows point in the same direction. Name each of them something appropriate. If you do not name them, Gambit will name them for you.
7. Close the *Create Straight Edge* window.

Generate a face from the edges

1. Under *Operation*, Geometry Command Button-Face Command Button-R-Form Face-Wireframe.
2. It is wise to select the edges *in order* when creating a face from existing edges. (I like to select them in mathematically positive counterclockwise order). Click in the Edges box to highlight it.
3. Select all four edges. These edges outline the “wireframe” of a closed face.
4. In *Create Face From Wireframe*, type in a label for this face if desired (“domain” is suggested), and Apply. If all went well, a blue outline of the face should appear on the screen; this is a face, which is now ready to be meshed.

Link the left and right edges so that they can be periodic boundary conditions

1. Under *Operation*, Mesh Command Button-Edge Command Button-Link/Unlink Edges.
2. Select *both* the left and right edges. Make sure that the Periodic option is turned on, and Apply.

Specify the boundary types (boundary conditions) on all edges

1. Under *Operation*, Zones Command Button-Specify Boundary Types Command Button.
2. Select the bottom edge. Name it “bottom_wall” or something equally descriptive. The default boundary type should be WALL, which is what we want, so Apply.
3. Repeat for the top wall, calling it “top_wall” or something else of your choosing.
4. Select *both* the left and right edges. Change the boundary type to PERIODIC, and name it “periodic”. Apply.
5. Turn on the Show labels option to double check that the boundary type on all four edges have been specified.

Generate the mesh

1. Under *Operation*, Mesh Command Button-Edge Command Button, and then Mesh Edges.
2. The window that should pop up is the *Mesh Edges* window. If not, Mesh Edges under *Edge* again.
3. Select the left edge. Make sure that the Pick with links option is on [should be on by default]. With this option turned on, Gambit will create identical node distributions on the left and right edges even though you pick only one of them; otherwise the periodic boundary condition won't work.
4. Click on the box called Interval size and change it to Interval count. To the left of that box, enter the number 10 (10 intervals along these edges). Apply. Blue circles should appear at the nodes along both the left and right edges.
5. Repeat for the bottom and top edges, but use 50 intervals.
6. Under *Mesh*, Face Command Button and then Mesh Faces.
7. In the *Mesh Faces* window, select the face [we have only one face in this computational domain]. The default settings are fine, so Apply. A nice structured rectangular mesh should appear.

Write out the mesh file and exit Gambit

1. In the main Gambit window, File-Export-Mesh. Turn on the 2-D option, and Accept. (The default file name can be changed here if desired, but the default name is acceptable in this case.)
2. The *Transcript* (at lower left) informs you when the mesh is written correctly to the file. If there is an error here, you must have done something incorrectly. You may need to re-do the mesh or the boundary conditions. **[If all else fails, start all over from the beginning.]**
3. Exit Gambit: File-Exit-Yes [you want to save one last time].
4. The mesh file should now be ready for use by Fluent. You may move on to the next learning module, [Fluent-2-D incompressible Couette flow](#), which describes how to run this case with the CFD solver, Fluent.