FLORIDA INSTITUTE OF TECHNOLOGY
MECHANICAL AND AEROSPACE ENGINEERING DEPARTMENT

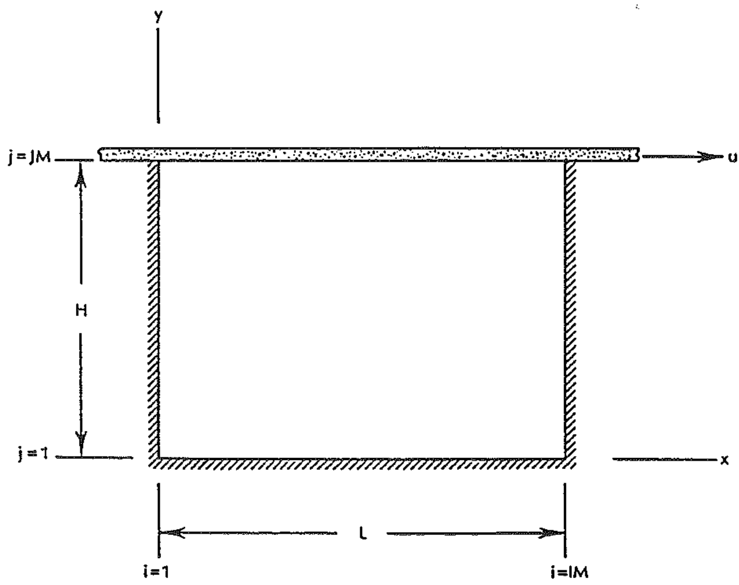
MAE 5150-E1: Computational Fluid Dynamics

Fall 2017

Final Project

Due December 11, 2017 @ 5:00pm

Consider the following figure:



The left, right, and bottom surfaces are stationary, while the top plate moves to the right with velocity $u = u_\infty$. This is known as *cavity-driven flow*, or simply *cavity flow*. For this project, you will write a code that will determine the laminar velocity field within the cavity. The equations to be solved are Eqns. (8-86, 8-87, and 8-88) in your textbook. Note that Eqn. (8.86) is the Poisson equation for pressure, and that

$$D = \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}$$

is called the *dilatation.*

These equations will be solved sequentially (not simultaneously) on a collocated mesh (not staggered). That is, for each time step, you will use Line Gauss-Seidel to iterate a solution to the pressure equation, followed by an explicit scheme to update the *x*- and *y*-components of velocity (which will use your pressure equation solution).

Assigned 10/26/17

Step 1: Write out on paper the numerical schemes for each equation.

- Because the specific numerical schemes you must use are not in your book, you must develop them.
- Poisson equation for pressure (Eqn. 8-86): Use Line Gauss-Seidel. You used Line Gauss-Seidel in a previous assignment, so you should be able to adapt the equation from that assignment for use in this project. (Be mindful of the RHS term.) Use second-order central differencing for all second-order derivatives, and a first-order forward difference for the time derivative. See page 335 of your text as a guide for how to discretize the right-hand side, but keep in mind that the particular scheme on that page is for a staggered mesh. You will be best served if you write your numerical scheme to utilize Line Successive Over-Relaxation (LSOR). Your code will run much faster if you do.
- Dilitation: The pressure equation includes the dilatation, which means you must find its value at each mesh node. Use second-order central differencing and the definition above to find its value. To find the dilatation for a wall nodal point, use second-order forward or backwards differencing, as appropriate, instead.
- Momentum equations (Eqns. 8-87 and 8-88): Here, you will use a first-order forward difference for the time derivatives and explicit first-order upwinding for the convective derivatives. Continue to use central differencing for the second-order derivatives. See Eqns. (8-80 and 8-81) for a guide on how to discretize these equations (again keeping in mind that these equations are discretized for a staggered mesh), and Eqn. (8.96) and the surrounding text about how to incorporate upwinding. Remember, you do not know in advance whether the velocity will be positive or negative at a particular node.

Step 2: Using the numerical schemes developed in Step 1, write a code to solve cavity-driven flow.

- Your mesh should be 81 x 81 nodes (*IM* x *JM*) with $\Delta x = \Delta y = 0.00625$. The plate moves to the right with a non-dimensional velocity of $u = 1$. The Reynolds number is $Re = 1000$. For each time step, you should iterate your Line Gauss-Seidel solution for the pressure to an error less than *errormax* = 0.001.
- Assume the top and bottom surfaces are held at a constant (non-dimensional) pressure of 3350. For the left and right surfaces, assume $\partial p/\partial x = 0$. Properly embed boundary conditions accordingly.
- Run your code for 15,000 time steps with a $\Delta t = 0.003$ (non-dimensional).
- For the Line Gauss-Seidel, if you decide to use LSOR, the optimal value for the relaxation parameter is 1.315. This should keep the number of iterations down to approximately 95-120 each time step.

Step 3: Create a streamline plot of the flow.

- Once you have your solution, use MATLAB or other plotting software to create a streamline plot of the flow. To do this, you will need to load *x*, *y*, *u*, and *v* variable data into the plotting software.

- If you use MATLAB, the following will be of use:

  streamline (x, y, u, v, SX, SY, [*step-size*, *max vertices*])

  where x, y, u, v are 81 x 81 matrices. SX and SY are matrices of data points of where to start each streamline. To determine these, I would recommend the following commands:

  ```
  sx=[0.0:0.05:0.5];
  sy=[0.0:0.05:0.5];
  [SX, SY] = meshgrid (sx, sy);
  ```

  Step-size is the length of each segment along the streamline, and *max vertices* is the number of segments. I would recommend *step-size* = 0.1 and *max vertices* = 2500. You may vary any of these parameters to obtain the best plot.

- If you do not use MATLAB for the plot, use your best judgment.
- Ensure your axes are scaled equally and have labels and titles. Your plot should also be titled. Use all proper plotting techniques. You will be graded on these.
- A vector plot may also be turned in if done legibly. By "legibly" I mean in such a manner that the arrows can be easily seen, but do not significantly overrun each other, and the primary flow structures can be viewed. Including a *proper* vector plot, though not required, will be viewed favorably in determining the project grade.

Step 4: To complete the project you must turn in:

- The numerical schemes you developed in Step 1, neatly written and formatted or (preferably) typed.
- The streamline plot and (optionally) the vector plot. Hardcopies only.
- A hardcopy of your code.
- An electronic copy of your code submitted via Canvas.

The project grade will depend upon:

- Developing the correct numerical schemes
- Obtaining the correct simulation results
- Using all proper numerical techniques including
  - correct implementation of Line Gauss-Seidel for the pressure equation
  - correctly embedding the appropriate boundary conditions
  - proper use of pressure solution in the momentum equations
- Showing correct results on plot(s) using all proper plotting techniques
- Turning in all required materials as described above