Fluid Mechanics – Tutorial Sheet #4

This tutorial sheet is graded. We will cover this tutorial on Monday, 5 September. If you need any help drop by PAP-05-13 and ask for Julien or Milad or send an email to JULIEN001@e.ntu.edu.sg.

You are requested to hand in your work as a hardcopy latest at the beginning of the tutorial class. For more information on the submission rules refer to "Logistics and introduction" PowerPoint by Prof. Ohl.

---

1. The planar flow of a jet impinging on a wall could be approximated by the *planar stagnation flow* velocity field, as follows:

$$\boldsymbol{V} = u\hat{\boldsymbol{\imath}} + v\hat{\boldsymbol{\jmath}} \;, \quad u = ax \quad , \qquad v = -ay$$

a) Verify that this vector field satisfies conservation of mass for an incompressible fluid.
b) Find the stream function ψ for this flow field
c) Verify that this flow field is irrotational,

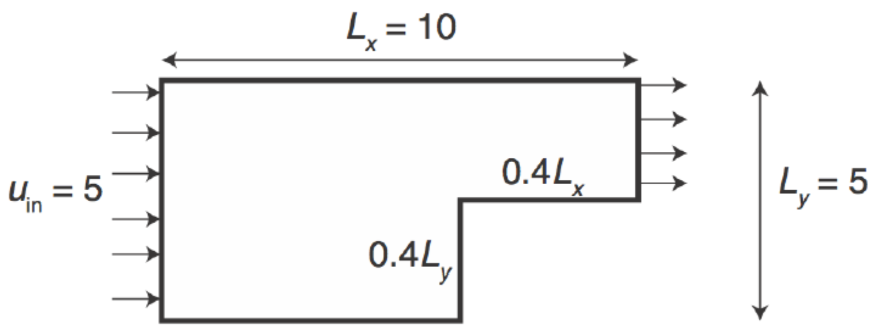
2. In this question, we compare two apparently similar planar flows:

solid body rotation vs. irrotational vortex.

a) Write down the velocity field for each of these flow fields in cylindrical coordinates: u_r, u_θ.
b) Calculate vorticity ω for each of these flow fields. Why is the vortex called irrotational?
c) From a textbook (e.g. Kundu), find the elements of the strain rate tensor in cylindrical coordinates: $S_{rr}, S_{r\theta}, S_{\theta\theta}$. What is the rate of linear and shear strain rate for each of these flow fields? Will a fluid element deform in both of those velocity field?
d) Find the stream function ψ for both of these flow fields in cylindrical coordinates. For which of the flow fields the velocity potential satisfies the Laplace equation and why?
e) Calculate the mass flow rate through the line connecting $(x = 1, y = 0)$ and $(x = 0, y = 1)$ for both of the flow fields. Use the stream function and be careful about the coordinates!

Hint: The different definitions and equations may be found in textbooks (e.g. Kundu) or in Prof. Ohl's lectures

3. In this question we solve the Laplace equation for the stream function using the finite difference numerical method. The computational domain is given as



The bottom and top boundaries are impermeable walls, to the left we have an inlet and the outlet is on the right.

For an incompressible and irrotational flow, the velocity field could be found by solving the Laplace equation for the stream function:

$$\nabla^2\psi(x,y) = \frac{\partial^2\psi}{\partial x^2} + \frac{\partial^2\psi}{\partial y^2} = 0$$

As explained in the lecture notes, the finite difference method is a scheme to approximate a system of differential equations with algebraic equations.

Using taylor approximation in a discretized Cartesian grid, the second derivatives could be approximated by:

$$\frac{\partial^2\psi}{\partial x^2} = \frac{\psi_{i+1,j} - 2\psi_{i,j} + \psi_{i-1,j}}{\Delta x^2} \quad , \quad \frac{\partial^2\psi}{\partial y^2} = \frac{\psi_{i,j+1} - 2\psi_{i,j} + \psi_{i,j-1}}{\Delta y^2}$$

In a uniform grid ($\Delta x = \Delta y$), if we insert the approximate derivatives into the governing equation ($\nabla^2\psi(x,y) = 0$), we obtain the algebraic equation that governs all the nodes and provides an approximate solution for the governing differential equation:

$$\psi_{i,j} = \frac{1}{4}(\psi_{i-1,j} + \psi_{i+1,j} + \psi_{i,j-1} + \psi_{i,j+1})$$

This approximation implies that the solution at any grid point is equal to the average of its neighbouring nodes. **Please go through week 4-5 lecture note and the iPython notebook "Numerical Solution to the Laplace Equation" to fully understand this method.**

Write a python program to solve the give problem using the following algorithm:

1) **Set up:** set your grid variables, e.g. $l_x, ly, \Delta x, \Delta y$, the grid coordinates of the step corners, etc.
2) **Initialization:** begin with a guess for $\psi_{i,j}$ at every grid point, simplest initial guess is $\psi_{i,j} = 0$ everywhere.
3) **Boundary conditions:** impose the boundary conditions on $\psi_{i,j}$. This means to override the values of the array $\psi_{i,j}$ on the boundaries (two impermeable walls, inlet, and outlet) with the values you obtain based on the boundary conditions. On an impermeable wall, ψ should be a

constant number, and in an inlet or outlet with uniform flow ψ varies linearly. You can find the boundary conditions based on the definition of the stream function.

4) **Update loop:** using the finite difference approximation for $\psi_{i,j}$ based on its neighbours, update the value for $\psi_{i,j}$ starting from the initial guess. If n represents the update loop counter, you can use the value from the previous values at neighbouring (ψ^n) nodes to calculate the new values (ψ^{n+1}).

$$\psi_{i,j}^{n+1} = \frac{1}{4}(\psi_{i-1,j}^n + \psi_{i+1,j}^n + \psi_{i,j-1}^n + \psi_{i,j+1}^n)$$

   In each update the boundary conditions have to be imposed again, therefore the update loop goes through these steps: **impose boundary conditions, update ψ, replace old ψ with new ψ, repeat.**

5) **Convergence:** the numerical method has converged to a solution if the value of ψ does not change considerably after each update; i.e. $|\psi_{i,j}^{n+1} - \psi_{i,j}^n| < error$.

Start with a grid of 21×11 and go up 101×51. To visualize the flow field, plot the contours of the stream function, and compare the results for the coarse and fine grid. Do you have any ideas to make the solver faster?