Posted 10/27/2025

AOE/ME 4434/5434 (Adv) Introduction to CFD
Fall 2025
Instructor: Dr. Chris Roy

Semester Project
Officially Due Wednesday, Dec. 10, 2025 at 11 pm ET
(Note: you can turn it in as late as Friday Dec. 12 at 11pm without penalty)

This is a two-person team project, although you may choose to work as an individual if you prefer. Teams must be composed of students with the same course number (i.e., both students must be in the graduate course AOE/ME 5434 or both must be in the undergraduate course AOE 4434). You may select your own teams, just enter your teams into one of the available groups on Canvas. If you have trouble finding a teammate, let me know and I will help you find one. If you decide to work alone, please place yourself into Group #40 or higher. Submit both a PDF of you project write-up and your original CFD code (*.f90, *.c, *.cpp, *.m, or *.py) to your chosen group #. If you have lot of program files, you may compress them and upload as a *.zip file.

Write a CFD code to solve for the flow in a square lid-driven cavity with the top wall moving at velocity U_{lid}. Use the 2D incompressible Navier-Stokes equations with time derivative preconditioning (assuming constant temperature and viscosity) given by

$$\frac{1}{\beta^2}\frac{\partial p}{\partial t}+\rho\frac{\partial u}{\partial x}+\rho\frac{\partial v}{\partial y}=S$$
$$\rho\frac{\partial u}{\partial t}+\rho u\frac{\partial u}{\partial x}+\rho v\frac{\partial u}{\partial y}+\frac{\partial p}{\partial x}=\mu\frac{\partial^2 u}{\partial x^2}+\mu\frac{\partial^2 u}{\partial y^2}$$
$$\rho\frac{\partial v}{\partial t}+\rho u\frac{\partial v}{\partial x}+\rho v\frac{\partial v}{\partial y}+\frac{\partial p}{\partial y}=\mu\frac{\partial^2 v}{\partial x^2}+\mu\frac{\partial^2 v}{\partial y^2}$$

where S is an artificial viscosity term given by

$$S=-\frac{|\lambda_x|_{\max}C^{(4)}\Delta x^3}{\beta^2}\frac{\partial^4 p}{\partial x^4}-\frac{|\lambda_y|_{\max}C^{(4)}\Delta y^3}{\beta^2}\frac{\partial^4 p}{\partial y^4}$$

where $|\lambda_x|_{\max}$ is the magnitude of the largest eigenvalues in (x, t) space, $|\lambda_y|_{\max}$ is the magnitude of the largest eigenvalues in (y, t) space, and $C^{(4)}$ is a constant that generally lies in the range

$$\frac{1}{128}\le C^{(4)}\le\frac{1}{16}.$$

In the above equation, β is the time-derivative preconditioning term given by

$$\beta^2=\max\left(u^2+v^2,\kappa U_{lid}^2\right)$$

where κ can range from 0.001 to 0.9.

Use the simple explicit method (i.e., a point Jacobi method) with second-order accurate central differences to advance the discrete equations in pseudo-time until you reach the steady-state solution. For students enrolled in the graduate course (AOE/ME 5434), also implement an explicit (point) symmetric Gauss-Seidel scheme. Monitor iterative convergence using the *steady-state iterative residuals* (i.e., the steady portion of your discretization evaluated all at the same time level). These should be plotted on a semiology scale with relative reduction of the iterative residual norms on the (log) y axis and iteration number on the (standard) x axis. A relative iterative convergence (i.e., the ratio of iterative residuals to initial iterative residuals at step 1) of *at least* 8 orders of magnitude is strongly recommended. Use the stability criteria we discussed in class to determine the time step; you may choose local or global time stepping, but local time stepping is recommended (i.e., take the largest allowable time step at each node).

The Fortran, C, C^{++}, MATLAB, and Python code templates you will be given have the capability to run manufactured solutions (by setting the *imms* input flag to one). Perform a code verification study by computing the discretization error norms ($|\epsilon_h|$, where for example $\varepsilon_h = u_h - \tilde{u}$) for the manufactured solution case on a series of systematically-refined meshes. Use a Reynolds number of 10, compute the observed order of accuracy, and show that it approaches second-order with mesh refinement.

Results

Run baseline cases at Reynolds number of 100 with the following conditions:

$$\mathrm{Re} = \frac{\rho U_{lid} L}{\mu} = 100$$
$$0 \le x \le 0.05\, m$$
$$0 \le y \le 0.05\, m$$
$$L = 0.05\, m$$
$$\rho = 1.0\, kg/m^3$$
$$U_{lid} = 1.0\, m/s$$

Note that the Reynolds number will determine the viscosity that you need to use. Examine the effects of mesh size, the κ parameter in the time derivative preconditioning, and the $C^{(4)}$ constant in the artificial viscosity. I recommend that the coarsest mesh you use be 33×33 nodes (starting with $i = j = 1$ at the lower left-hand corner where $x = y = 0$). The code is set up to perform pressure re-scaling to enforce a reference pressure at the center of the cavity, with the reference pressure p_{ref} = set to 0.801333844662 N/m^2 (consistent with the manufactured solution). For those you taking the graduate course (AOE/ME 5434), also run Reynolds numbers of 500 and 1000 (note, these cases may require finer meshes to be stable). You will be supplied with numerical solutions from the literature (and possibly experimental data) for comparison purposes.

A formal project report is required which should include sections on the theory (governing equations and boundary conditions), discretization, stability, artificial viscosity, iterative convergence, code verification, discretization error estimation via Richardson extrapolation (*for the lid-driven cavity case, not the manufactured solution*), and flow analysis (again, for the lid-driven cavity case). While the choice of programming languages is up to you, I will be giving you the code structure in Fortran 95/2003, C, C^{++}, MATLAB, and Python, and thus recommend that you write your program in one of these languages. You will be required to complete a number of subroutines as part of this project. Please note that if you choose to use MATLAB or Python, it will likely run significantly slower than the Fortran/C/C^{++} codes. Include your CFD code as a separate file in your submission and also include it as an appendix in your report, which must be submitted in PDF format. Compare your code results to those that you obtain using a commercial or research CFD code. You will be given access to ANSYS/Fluent, which we will be discussing later in course. Limit your report (excluding appendices) to 15 pages. Space will be at a premium, so think carefully about how to include the key important results and use appendices as necessary. A report template will also be made available (highly encouraged although not required).

Put your code under version control using Git, TortoiseGit, or similar version control software. Check your coding revisions into your repository frequently to ensure that you do not lose any of your work and to allow you to go back to previous versions of your code. Put a screen shot of a “diff” between two of your versions into the first appendix of your report (you can just use Alt-Print Screen or Function-Alt-Print Screen in Windows).

Note: The honor code will be strictly enforced on this project. Any individuals or teams caught cheating (e.g., using parts of someone else’s program) will be reported to the Honor System. <u>This is easy for us to check</u>.

Some items that should be included in your report follow below (note: most of these should be in the main body of your report or, at a minimum, in the first few appendices).

1) Residual history for conservation of mass, x-momentum, and y-momentum:

- Residual comparison between different relaxation methods (PJ, SGS): the SGS is not needed for undergraduates (AOE 4434)
- Residual comparison among different mesh sizes (at least two, e.g., 65x65 and 129x129)

2) Comparison of MMS solutions and your solutions of *u* at a reasonable size mesh.

3) Plots of norm of DE (log-log scale) and observed or accuracy $\hat{p}$ (semilogx scale) versus a representative node spacing (e.g., sqrt(#nodes)) for the MMS case.

4) Contours of solutions of *p*, *u*, and *v* (including streamlines for *u*) at Re = 100.

5) Effects of mesh size (contour plot of *u* with streamlines) at Re = 100.

6) Estimate of discretization error or discretization uncertainty using Richardson extrapolation (x-y plot of u velocity for a vertical line through the center of the channel).

7) Effects of Re via contour plot of *u* and streamlines for your choice of mesh resolution (AOE/ME 5434 only).

8) Effects of kappa (κ) on the iterative residual for the cavity case (e.g., at 65x65 and Re=100; note, this should be combined with #1 above).

9) Effects of $C^{(4)}$ on the MMS solution: plot DE of pressure versus *x* location ($0<x<0.01$ m) at *y* near, but not at, 0.05 m (top of cavity). Refer to the plot in the lecture notes for an example.

10) Comparison with literature, for example *u* or *v* velocity profile along the centerline of the cavity and compare it with the results from the papers.

11) Use ANSYS/Fluent, openFOAM, or another CFD code to compare to some of your results.