Michael D. Walker
APC523: Numerical Algorithms for Scientific Computing
March 23, 2024

Final Project Proposal:
Comparing Iterative Methods on Pressure Solver Performance
for Incompressible Flow Simulation

1. Introduction

Computational fluid dynamics provides effective means for the rapid evaluation of design scenarios and can produce significant detail of the fundamental physical processes in a flow situation [5, 6]. As a result, there is great motivation to develop low cost tools that enhance education and design. Experimental and prototype setups are often expensive and test conditions are difficult to replicate; numerical solutions provide a promising and reliable alternative. We propose to develop an incompressible flow solver of the Navier-Stokes formulations using the explicit Euler method, applied to 2-D domains of several test cases. Further, we will explore different iterative methods (Jacobi, Gauss-Seidel, Successive Over-Relaxation) to resolve the pressure term. Test cases will consider a range of length and velocity scales (Reynolds numbers) and error tolerance. Time permitting, we will implement additional numerical schemes and software performance engineering to improved speed, accuracy, and stability.

2. Governing Equations in Fluid Mechanics

Fluid flows present a non-linear, multi-scale problem described by the inherently coupled system of Navier-Stokes equations. These differential equations define conservation of mass and momentum assuming the continuum hypothesis [1]. The conservation of mass principle (continuity equation) is written as

$$\frac{\partial \rho}{\partial t} + \frac{\partial(\rho u_j)}{\partial x_j} = 0 \ ,$$

where ρ is the density of the fluid and u_j is the velocity in the principal direction j.

Understanding the variation of velocities and scalars (e.g., ρ or T) in the flow at all conditions is necessary to describe the flow structure. The conservation of momentum is described by the Navier-Stokes transport equation

$$\frac{\partial(\rho u_i)}{\partial t} + \frac{\partial(\rho u_j u_i)}{\partial x_j} = -\frac{\partial P}{\partial x_i} + \rho g_i + \frac{\partial \tau_{ij}}{\partial x_j} \ ,$$

where u_i is the velocity in the principal direction i, P is the pressure at each point in the fluid, g_i is the body force in direction i, and τ_{ij} is the viscous shear stress. For constant-property Newtonian fluids, τ_{ij} is related to strain rate by

$$\tau_{ij} = \mu\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right) - \frac{2}{3}\mu\frac{\partial u_k}{\partial x_k}\delta_{ij} \ ,$$

where u_k is the velocity in the principal direction k, μ is the dynamic viscosity of the fluid, and δ_{ij} is the Kronecker delta function.

The conservation of energy equation exists in many forms, but is presented here in terms of enthalpy for an arbitrary number of chemical species [1] as

$$\frac{\partial(\rho h)}{\partial t} + \frac{\partial(\rho u_j h)}{\partial x_j} = \frac{\partial P}{\partial t} + \frac{\partial}{\partial x_j}\left(\lambda\frac{\partial T}{\partial x_j}\right) + \frac{\partial}{\partial x_j}\left(\rho \sum_{\alpha=1}^{N} D_\alpha\, h_\alpha \frac{\partial Y_\alpha}{\partial x_j}\right) \ ,$$

where T is the temperature of the fluid, λ is the thermal conductivity, Y_α is the mass fraction of species α, h_α is the absolute enthalpy of species α, N is the total number of species in the mixture, D_α is the diffusion coefficient for species α, where Fick's law for diffusion is assumed, and h is the mixture enthalpy $\sum_{\alpha=1}^{N} Y_\alpha\, h_\alpha$. This formulation neglects radiation, and viscous heating and acoustic interactions under a low-speed assumption [1]. For a homogeneous non-chemically reacting flow, it can be assumed $\alpha = 1$ and $Y_\alpha = 1$.

An accompanying appropriate equation of state is required to close this system of equations. This usually takes the form of the ideal gas law, $P = \rho R T / \overline{W}$, where R is the universal gas constant and $\overline{W}$ is the mean molecular mass of the mixture. This system is readily solved for laminar flows, but analytical solutions do not currently exist for unstable or turbulent flows. There are multiple strategies to model such flows by solving a form of these equations numerically.

3. **Explicit Euler Fluid Solver**

The Navier-Stokes equations represent a system of non-linear hyperbolic partial differential equations that can be closed with a continuity equation and some form of an equation of state relating the thermodynamic quantities. This project will be limited to a class of fluid problems that can be considered incompressible. With this assumption, it can be assumed there are no density fluctuations, so to solve the incompressible Navier-Stokes equation, a simple explicit Euler method can be used [2, 5]. This method is characterized by an update to the velocity for each time-step,

$$u_i^{n+1} - u_i^n = \Delta t \left[-\frac{\delta u_i u_j}{\delta x_j} + \nu \frac{\delta^2 u_i}{\delta x_j{}^2} - \frac{\delta P^n}{\delta x_i} \right]$$

where $\delta / \delta x$ is an approximation to the spatial derivative (a second-order central-difference for this implementation). To ensure that mass is conserved, the divergence of will be taken with $\delta u_i^{n+1} / \delta x_i = 0$ enforced. This requires that to ensure the continuity of the velocity at the next time-step, a Poisson equation for the pressure P^n must be solved at each time-step. The Poisson equation can be expensive to calculate, so different iterative approaches will be discussed in the next section.

With consideration for timing and scope, we will seek to implement several "enhancements"–numerical methods to improve speed (residual averaging, constant stagnation enthalpy, spatially varying time steps), accuracy (deferred corrections), and stability (Runge-Kutta multi-stage method).

4. **Pressure Solvers and Iterative Methods**

In this scheme [3, 4], the solution to the pressure term is both required to enforce conservation of mass and computationally expensive. To solve to the $\partial P / \partial x_i$ term, a surface integral around the faces of the two-dimensional grid cell can be taken on both sides of the equation,

$$\frac{\partial P}{\partial x}\Big|_{i+0.5,j} - \frac{\partial P}{\partial x}\Big|_{i-0.5,j} + \frac{\partial P}{\partial y}\Big|_{i,j+0.5} - \frac{\partial P}{\partial y}\Big|_{i,j-0.5} = \rho \Big[H_{x|i+0.5,j} - H_{x|i-0.5,j} + H_{y|i,j+0.5} - H_{y|i,j-0.5} \Big]$$

where $H_i^n \equiv -\delta u_i u_j / \delta x_j + \nu \delta^2 u_i / \delta x_j{}^2$. The values at the cell faces are then interpolated from the cell centered values, and the pressure gradient terms can be expanded using the second order central difference kernel, linear interpolation, and simplified. A no-slip condition (zero velocity) and zero pressure gradient boundary condition can be implemented at the walls.

4.1. **Jacobi Iteration.** To implement a Jacobi iterative scheme, a new pressure array is created that has its values set to the right hand side of the corresponding discretized pressure equation. The difference between the new pressure and old pressure at each location is calculated, and the maximum of this difference over the entire grid is taken to be the error of the iteration. The arrays are then swapped and the process is repeated until the error is less than the defined tolerance.

4.2. **Gauss-Seidel Iteration.** For a Gauss-Seidel iterative scheme, the process is very similar, however values to the left of the cell being updated are evaluated with the already updated value.

4.3. **Successive Over-Relaxation.** For Successive Over-Relaxation (SOR), the initial process similar to the Gauss-Seidel iteration, however there is an additional step where an over-relaxation parameter ω is defined and varied by the user to determine the optimal value for a particular set of equations.

5. Test Cases

We will seek to simulate a lid driven cavity flow problem. This consists of a box where the left, right and bottom walls are enforced with a no slip condition (velocity of the fluid at the wall is zero so that the velocity gradient is continuous). The top wall is then moved in either direction with a specified velocity and forces the fluid in the cavity to move. The velocity of the top wall is set to a modest Mach number Ma = 0.5 in an incompressible flow with initial pressure of one atmosphere and an initial temperature of 300K. In order to analyze the effects of the various pressure solvers on the performance of the entire incompressible solver, three cases will be considered.

5.1. **High Tolerance, Low Reynolds Number.** The high tolerance, low Reynolds number case is designed to demonstrate (without a large amount of computational complexity) that each of the pressure solvers can converge to a solution. The low Reynolds number, Re = 100, is important because with viscosity, the system reaches a steady state where the solution to the pressure equation does not change much between iterations.

5.2. **Low Tolerance, Low Reynolds Number.** The low tolerance, low Reynolds number case is designed to be a direct comparison to the previous case, because with the lower error tolerance on the pressure solver, the difference in performances of each of the methods can be observed. This is relevant because with many of these simulations, it is important to have the pressure solved extremely accurately as it enforces mass continuity and thus physical accuracy. As mentioned above, the low Reynolds number allows the system to be relatively viscously dominated. It is also expected that for some of the methods, this simulation should take longer to run.

5.3. **High Tolerance, High Reynolds Number.** This final test case will determine the extent to which the pressure solver is sensitive to the flow behavior when the flow is no longer viscously dominated, and there are smaller scale motions present within the system. The Reynolds number will be increased to Re = 1000 which will show different flow characteristics than with a Reynolds number of 100. In addition, with the SOR solver, varying values of ω will be explored to determine the optimal value for this specific set of equations.

6. Personal Benefits

This project will allow us to develop further proficiency multiple aspects of the numerical method topics of this course. We work in a computational group (the Computational Turbulent Reacting Flow Laboratory, CTRFL), and will seek to incorporate any lessons learned. While not relevant to this project, the course content in massively parallel architectures was also extremely relevant to our research.

References

[1] R.S. Cant and E. Mastorakos. *An Introduction to Turbulent Reacting Flows.* Imperial College Press, 2007.
[2] C. A. J. Fletcher. *Computational Techniques for Fluid Dynamics: Fundamental and General Techniques.* Cambridge Aerospace Series. Springer-Verlag Berlin Heidelberg, 1998.
[3] S. T. Fush. VIPS: Varying Incompressible flow Pressure Solver. https://github.com/trevorfush/vips, 2023.
[4] T. Hynes and P. G. Tucker. *Course Lecture Slides.* Computational Fluid Dynamics Part IIB 4A2. University of Cambridge Department of Engineering, Michaelmas 2018.
[5] B. E. Launder and D. Spalding. The numerical computation of turbulent flows. *Computer Methods in Applied Mechanics and Engineering,* 3(2):269–289, 1974.
[6] T. Saad. *Turbulence Modeling For Beginners.* University of Tennessee Space Institute, 2011.