Michael D. Walker (mw6136)
MAE557: Simulation and Modeling of Fluid Flows
October 24, 2025

Mini-Project 2 Report

Disclaimer: I collaborated with Philip Satterthwaite and Michael Schroeder. I received assistance from Trevor Fush and Israel Bonilla. I used Google AI Studio (Gemini 2.5 Pro) for debugging. Contrary to the instructions from the project workshop, I did *not* have fun.

Abstract

A numerical solver was developed to simulate two-dimensional, unsteady, compressible viscous flow in a lid-driven cavity with a sinusoidally oscillating top wall. The solver is based on a finite-difference discretization of the compressible Navier-Stokes equations. The governing equations were formulated in conservative form to ensure the numerical conservation of mass, momentum, and energy. A second-order central difference scheme was employed for spatial discretization, and an explicit Forward Euler method was used for time integration. This report details the numerical methodology, presents a verification study of the solver's spatial and temporal convergence, and investigates the flow behavior across a range of Mach numbers. Additionally, the thermodynamic effect of adiabatic versus isothermal wall boundary conditions is explored. The solver was successfully verified, demonstrating the expected first-order temporal and second-order spatial accuracy. The study confirmed that the computational efficiency of the compressible flow solver increases significantly with the Mach number.

Introduction

This project concerns the simulation of a compressible, viscous fluid within a two-dimensional square cavity $L = 1$ [m]. The flow is driven by the top wall, which moves tangentially with a time-dependent sinusoidal velocity profile, while the other three walls remain stationary. Zero-slip/zero-penetration boundary conditions are established on the four walls and lid. This configuration, a variation of the classic lid-driven cavity problem, serves as a test case for numerical methods in fluid dynamics, incorporating unsteadiness, viscous effects, and compressibility.

The gas is calorically perfect air $\gamma = 1.4$ with constant specific heats. The initial pressure and temperature are uniformly $P = 1$ [bar] and $T = 300$ [K], respectively, with a specific gas constant of $R = 287$ [J/kg-K], so $c_P = \gamma R/(\gamma - 1)$. This initial temperature is used to define the reference speed of sound a_0. The walls are isothermal (Dirichlet boundary condition $T_{\text{wall}} = 300$). The kinematic viscosity ν and thermal diffusivity $\lambda/\rho c_P$ are constant, but the dynamic viscosity $\mu = \rho\nu$ and the ratio λ/c_P vary with density ρ. The properties are defined non-dimensionally by the Reynolds number $\text{Re} = |u|_{\text{wall}}L/\nu = 100$, Mach number $\text{Ma} = |u|_{\text{wall}}/a_0 = 0.025$, Prandlt number $\text{Pr} = \rho\, c_P\, \nu/\lambda = 0.7$, and lid oscillating frequency $L\sqrt{\omega/2\nu} = 1$.

A.   Governing Equations for Compressible Flow

The governing equations for this problem are the two-dimensional, unsteady, compressible Navier-Stokes equations. To ensure that mass, momentum, and energy are conserved at the discrete level, the equations were implemented in their conservative form. This choice is crucial for numerical stability and for accurately capturing sharp gradients or discontinuities. The conservative form is expressed as:

(1)
$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{F}}{\partial x} + \frac{\partial \mathbf{G}}{\partial y} = 0$$

where t is time, x and y are the spatial coordinates, $\mathbf{U}$ is the vector of conserved variables, and $\mathbf{F}$ and $\mathbf{G}$ are the conservative flux vectors in the x and y directions, respectively. These vectors are defined as:

$$(2) \qquad \mathbf{U} = \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ \rho e_t \end{bmatrix}, \quad \mathbf{F} = \begin{bmatrix} \rho u \\ \rho u^2 - \sigma_{xx} \\ \rho uv - \sigma_{xy} \\ (\rho e_t - \sigma_{xx})u - \sigma_{xy}v - k\frac{\partial T}{\partial x} \end{bmatrix}, \quad \mathbf{G} = \begin{bmatrix} \rho v \\ \rho uv - \sigma_{yx} \\ \rho v^2 - \sigma_{yy} \\ (\rho e_t - \sigma_{yy})v - \sigma_{yx}u - k\frac{\partial T}{\partial y} \end{bmatrix}$$

Here, ρ is the fluid density, u and v are the velocity components, e_t is the total energy per unit mass, T is the temperature, and k is the thermal conductivity. The total stress tensor, σ_{ij}, is given by $\sigma_{ij} = -P\delta_{ij} + \tau_{ij}$, where P is the pressure, δ_{ij} is the Kronecker delta, and τ_{ij} is the viscous stress tensor.

B. Finite Difference Operators for Spatial Discretization

A conservative finite difference approach was chosen for the spatial discretization. The operators were applied to the governing equations in their conservative form (Equation 1) to ensure that mass, momentum, and energy are conserved at the discrete level. Second-order centered difference operators were selected for this task. The computational domain was divided into a grid of uniform cells. It should be noted that on a structured grid, this method is mathematically similar to a cell-centered finite-volume method. This scheme was chosen for two primary reasons:

1) **Conservation and Accuracy:** By evaluating the conservative fluxes ($\mathbf{F}$ and $\mathbf{G}$) at the cell faces and then differencing them, the method ensures secondary conservation. This means the flux leaving one cell face is precisely the same as the flux entering the adjacent cell, which is critical for numerical stability and accuracy. The centered nature of the operators provides second-order accuracy for smooth flows.
2) **Non-dissipative Nature:** Unlike upwind schemes, central differencing does not introduce artificial viscosity, which can improperly damp physical oscillations and reduce the effective Reynolds number of the simulation.

The resulting semi-discrete update for a cell centered at (i, j) is formulated as:

$$(3) \qquad \frac{d\mathbf{U}_{i,j}}{dt} = -\left[\frac{\mathbf{F}_{i+\frac{1}{2},j} - \mathbf{F}_{i-\frac{1}{2},j}}{\Delta x} + \frac{\mathbf{G}_{i,j+\frac{1}{2}} - \mathbf{G}_{i,j-\frac{1}{2}}}{\Delta y} \right]$$

where the fluxes are evaluated at the cell faces (e.g., $i + \frac{1}{2}, j$). This flux-differencing approach requires only a single layer of ghost cells to enforce the boundary conditions. The fluxes at the cell faces were approximated using second-order centered difference operators. For example, the velocity derivatives required for the viscous stress tensor at the cell faces are calculated as:

$$(4) \qquad \left.\frac{\partial u}{\partial x}\right|_{i+\frac{1}{2},j} = \frac{u_{i+1,j} - u_{i,j}}{\Delta x} \qquad\qquad \left.\frac{\partial u}{\partial y}\right|_{i-\frac{1}{2},j} = \frac{u_{i-\frac{1}{2},j+\frac{1}{2}} - u_{i-\frac{1}{2},j-\frac{1}{2}}}{\Delta y}$$

This approach ensures secondary conservation, meaning that fluxes leaving one cell enter the adjacent cell, maintaining the conservation property at the discrete level.

C. Temporal Scheme and Solution Method

For time integration, an explicit Forward Euler scheme was selected. This is a single-step, first-order accurate method. The fully discrete update equation is:

$$(5) \qquad \mathbf{U}_{i,j}^{n+1} = \mathbf{U}_{i,j}^{n} - \Delta t \left[\frac{\mathbf{F}_{i+\frac{1}{2},j}^{n} - \mathbf{F}_{i-\frac{1}{2},j}^{n}}{\Delta x} + \frac{\mathbf{G}_{i,j+\frac{1}{2}}^{n} - \mathbf{G}_{i,j-\frac{1}{2}}^{n}}{\Delta y} \right]$$

where the superscript n denotes the time level. The Forward Euler scheme was chosen primarily for its simplicity of implementation. While higher-order explicit methods (like Runge-Kutta) or implicit methods could offer better accuracy or stability, the Forward Euler method provides the most straightforward implementation, which is advantageous for initial solver development and verification. While implicit methods allow for larger time steps, they require solving a system of equations at each step, which is more

complex to code. The explicit approach has a lower computational cost per time step but is subject to stringent stability constraints (limited conditional stability), which is addressed in the next section.

D. Expected CFL Limits

Explicit time-marching schemes are conditionally stable, with the maximum allowable time step, Δt, being limited by the Courant-Friedrichs-Lewy (CFL) condition. For the 2-D compressible Navier-Stokes equations, the time step is constrained by both convective and viscous phenomena. The stability condition can be approximated as:

$$
(6) \qquad \Delta t \leq \text{CFL} \cdot \mathbf{min} \left(\frac{\Delta x}{|u| + a_0}, \frac{\Delta y}{|v| + a_0}, \frac{\rho(\Delta x)^2}{2\mu}, \frac{\rho(\Delta y)^2}{2\mu} \right)
$$

where a_0 is the speed of sound and μ is the dynamic viscosity. For a Forward Euler scheme with second-order central differencing, the theoretical CFL limit is typically on the order of 1 for purely hyperbolic problems. However, the presence of viscous terms adds a more restrictive diffusion limit. Epirical testing for this specific incompressible flow problem revealed that a CFL number greater than 0.1 led to spurious oscillations and simulation failure. The most stable results were obtained with a CFL number of approximately 0.01, which allowed for a run time up to $\omega t = 10$. This value is lower than the theoretical ideal but is not uncommon for explicit schemes applied to complex flows where multiple stability constraints interact.

E. Algorithm in Pseudo-Code

The simulation proceeds by marching forward in time from a specified initial condition. The algorithm for a single time step is as follows. `while` $t < t_{\max}$:

1) **Enforce Boundary Conditions:** Update the values in the ghost cells based on the physical boundary conditions (no-slip, isothermal/adiabatic walls, oscillating top lid) at the current time t^n.

2) **Compute Primitive Variables:** From the vector of conserved variables $\mathbf{U}^n$, calculate the primitive variables (ρ, u, v, p, T) for each cell.

3) **Calculate Face Values:** Interpolate primitive variables to the cell faces. For a second-order scheme, this is a simple average of adjacent cell-centered values.

4) **Compute Gradients:** Calculate the velocity $\partial u_i / \partial x_i$ and temperature $\partial T / \partial x_i$ gradients at the cell faces using centered differences.

5) **Compute Fluxes:** Using the face values of the primitive variables and their gradients, compute the viscous stress tensor τ_{ij} (and total stress tensor σ_{ij}) and the heat flux. Assemble the full conservative flux vectors $\mathbf{F}^n$ and $\mathbf{G}^n$ at all cell faces.

6) **Update Conserved Variables:** Apply the Forward Euler update (Equation 5) to compute the new conserved variable vector $\mathbf{U}^{n+1}$ for all interior cells.

7) **Advance Time:** Increment time $t^{n+1} = t^n + \Delta t$. Repeat the loop until the final simulation time is reached.

F. Implementation and Convergence Analysis

To prove the correctness of the implementation, the solver's order of accuracy was tested in both space and time. A reference solution was generated on a highly refined grid, and the L_2 norm of the error was computed for a series of coarser grids and larger time steps.

Spatial Convergence The spatial accuracy was assessed by running the simulation on a series of progressively refined grids while keeping the time step sufficiently small to ensure that spatial errors dominated. Figure 1 shows a log-log plot L_2 norm of error versus grid spacing, Δx. The norm was calculated using a reference value $u^\star$ from the highest resolution case. The measured slope of the error curve is approximately 2.33. This is close to the theoretical value of 2 for a second-order accurate spatial discretization, thus verifying the correct implementation of the central difference scheme.

Temporal Convergence A similar analysis was performed to assess temporal convergence. The grid was held fixed at a high resolution, and the time step Δt was systematically reduced. Figure 1 shows a log-log plot of the error versus the time step. The slope of the line is slightly less, but near 1.0, which is consistent with the theoretical first-order accuracy of the Forward Euler scheme. This result confirms that the temporal integration is implemented correctly and that the solver is performing as designed.
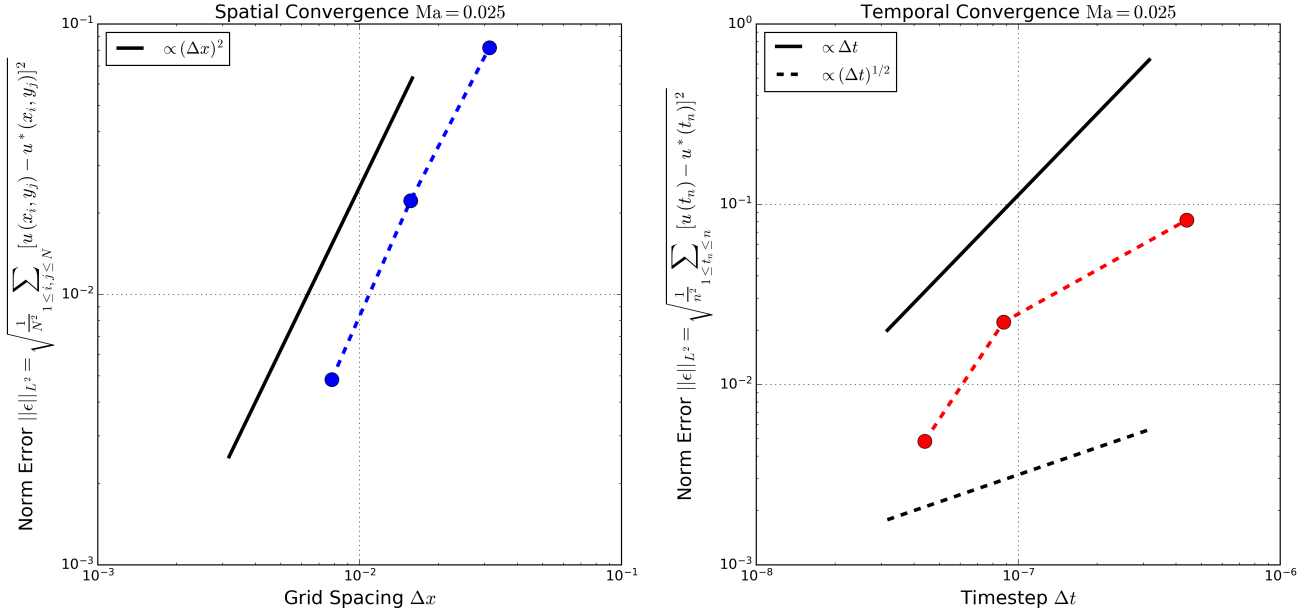

FIGURE 1. (left)Spatial convergence of the flow solver. The slope of approximately 2.33 confirms the expected second-order accuracy. (right) Temporal convergence of the flow solver. The slope of approximately 1.0 confirms the expected first-order accuracy of the Forward Euler scheme.

G. Mach Number Study

The simulation was run for a range of Mach numbers (0.025, 0.05, 0.1, 0.2, 0.4, 0.8) up to a non-dimensional time of $\omega t = 10$. The Reynolds number (i.e., $\mathrm{Re} = |u_\mathrm{wall}|L/\nu = 100$) and non-dimensional frequency (i.e., $L\sqrt{\omega/2\nu} = 1$) were held constant by increasing the wall velocity magnitude $|u_\mathrm{wall}|$ and frequency of the oscillation ω, and decreasing the domain size L accordingly.

Qualitative Flow Features Figure 2 compares the velocity magnitude fields for the different Mach numbers. The flow in all cases is stable and exhibits the expected structure, characterized by a primary vortex driven by the lid. The overall flow pattern is qualitatively similar across the cases, with the primary difference being the magnitude of the velocity, which scales directly with the Mach number as expected.

Computational Efficiency A key finding of this study was the significant increase in computational efficiency at higher Mach numbers. Ultimately compressible flow solvers are limited temporally by the sound speed in the medium, so at low mach numbers the timesteps are small relative to the speed of development of flow phenomena. This is a well-known characteristic of explicit compressible flow solvers. The stability of the scheme is limited by the speed of sound, a_0, which is much larger than the fluid velocity at low Mach numbers. Consequently, the time step Δt must be very small, and many iterations are needed to reach a given physical time. This highlights the "stiffness" problem associated with simulating low-Mach-number flows with a compressible formulation. As the Mach number increases, the fluid velocity becomes a more significant fraction of the sound speed. This allows for a larger Δt relative to the characteristic time scale of the flow's evolution, enabling the simulation to proceed more efficiently.
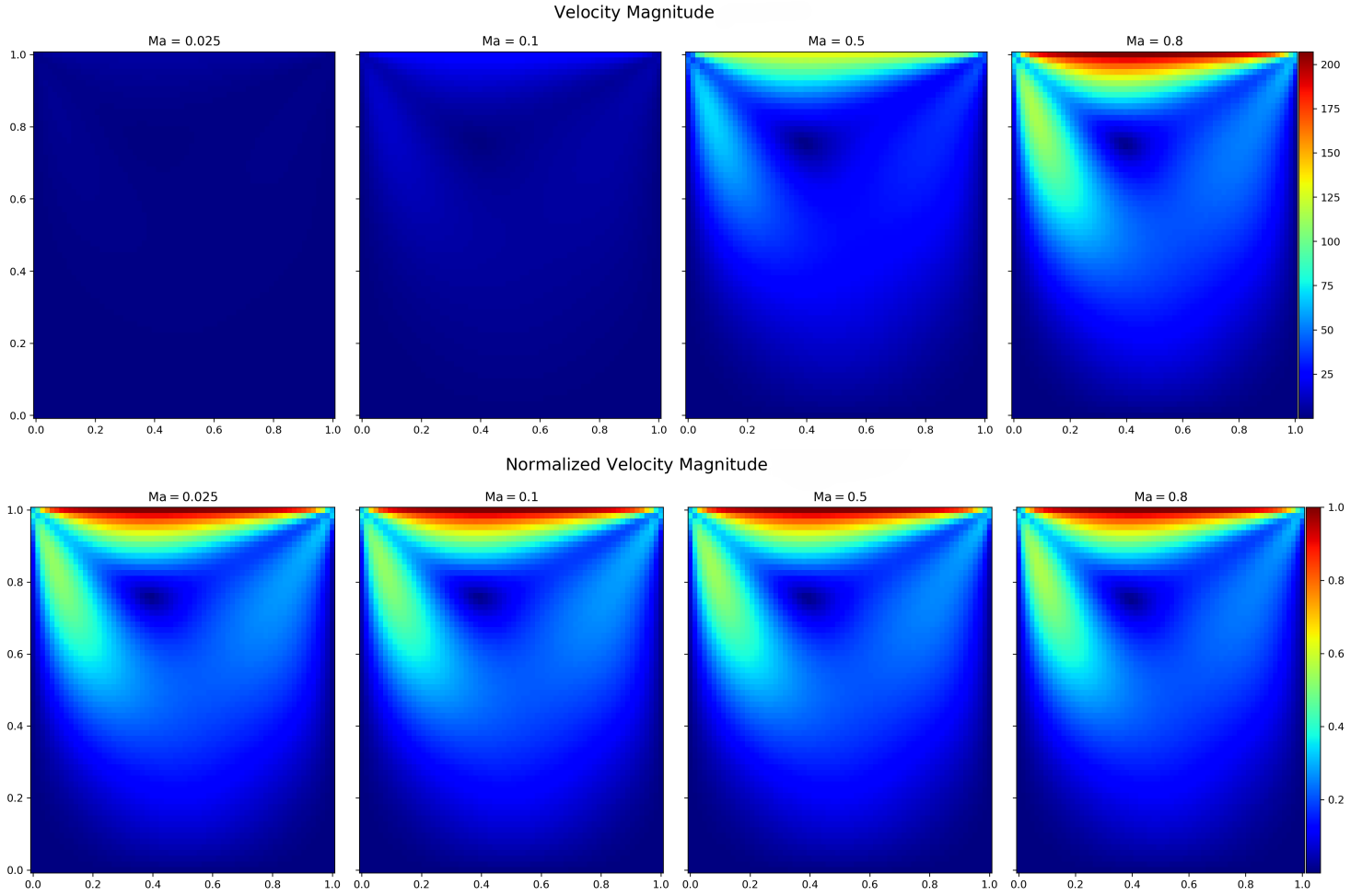
FIGURE 2. Velocity magnitude $|\mathbf{u}| = \sqrt{u^2 + v^2}$ and normalized velocity magnitude $|\mathbf{u}|/|\mathbf{u}|_{\max}$ at $\omega t = 10$ for varying Mach numbers. The flow structure is stable and qualitatively similar, with velocity magnitudes scaling with Mach number. Note that the domain size was varied to maintain constant Reynolds number and non-dimensional frequency.

H. Bonus: Adiabatic Wall Boundary Conditions

To investigate the effect of thermal boundary conditions, the isothermal walls ($T_{\text{wall}} = $ constant) were replaced with adiabatic walls ($dT_{\text{wall}}/dx_i = 0$), a Neumann boundary condition which prevents heat transfer out of the domain. In this scenario, the kinetic energy dissipated into thermal energy by viscosity is trapped within the cavity. This process, known as viscous heating, leads to a continuous increase in the fluid's internal energy and temperature.

As shown in Figure 3, the temperature in the cavity rises significantly above its initial value in just a short amount of time. This effect is more pronounced at higher Mach numbers. The rate of viscous dissipation is proportional to the square of the velocity gradients, so flows with higher velocities (i.e., higher Mach numbers) and stonger gradients generate heat much more rapidly. With no mechanism for this heat to escape, the temperature rise is substantial. At smaller Mach numbers, the rate of viscous heating is much lower, and the temperature rise over the same period is less pronounced.

Conclusion

A two-dimensional compressible flow solver was successfully developed using a conservative finite volume formulation, a second-order central difference scheme for spatial discretization, and a Forward Euler method for time integration. The solver's implementation was verified to be nearly second-order accurate

in space and nearly first-order in time.

The study of varying Mach numbers demonstrated that while the qualitative flow structure remained similar, the computational efficiency of the explicit solver increased markedly at higher Mach numbers. Furthermore, the investigation of adiabatic boundary conditions confirmed the physical principle of viscous dissipation, showing a significant temperature rise within the cavity, an effect that is amplified at higher Mach numbers.
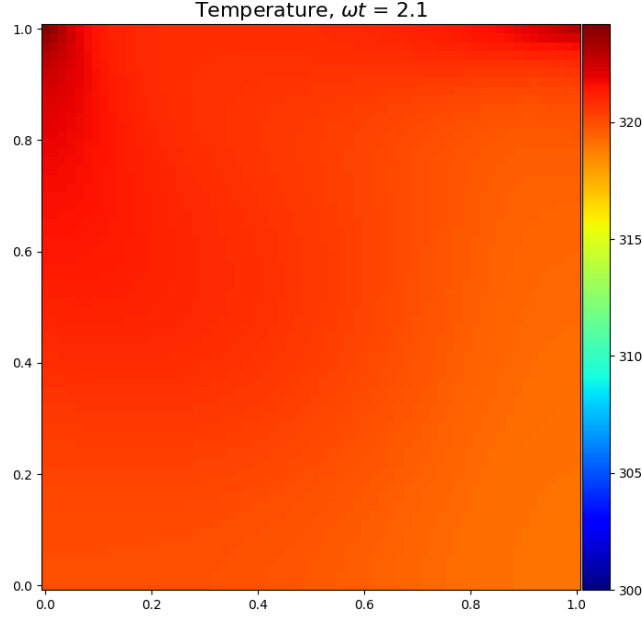

FIGURE 3. Temperature field at $\omega t = 2.1$ for a simulation with adiabatic wall boundary conditions, showing a significant temperature increase over $T = 300$ [K] due to viscous heating.