Computational Fluid Dynamics
Final Report

Piero Zappi

June 2024

1 Introduction

This report presents the study and the simulation of two classic fluid dynamics problems: the 2D lid-driven cavity flow and the 2D Poiseuille flow.
The lid-driven cavity problem was solved using two different approaches. First, a custom code was developed to solve the two dimensional incompressible Navier-Stokes equations using a finite difference-based method. Then, the same flow was simulated employing *OpenFoam*, an open-source CFD software. The obtained results were compared with those of *Ghia, Ghia and Shin* [1].
Finally, the Poiseuille flow was reproduced using *OpenFoam*; the results were validated against the analytical solution to ensure the accuracy and the reliability of the simulation.

2 Lid-driven cavity problem

The lid-driven cavity problem consists of a square cavity with a lid that moves at constant velocity, driving the flow inside the cavity. The fluid is assumed to be incompressible and the flow is assumed to be two-dimensional; the governing equations for this problem are the Navier-Stokes equations.
The two-dimensional incompressible Navier-Stokes equations in the conservative form, which consist of the continuity equation and the two momentum equations, are given by:

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0$$

$$\begin{aligned} \frac{\partial u}{\partial t} &= -\frac{\partial uu}{\partial x} - \frac{\partial vu}{\partial y} - \frac{\partial P}{\partial x} + \nu \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) \\ \frac{\partial v}{\partial t} &= -\frac{\partial uv}{\partial x} - \frac{\partial vv}{\partial y} - \frac{\partial P}{\partial y} + \nu \left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} \right) \end{aligned} \tag{1}$$

where $P = \dfrac{p}{\rho}$ is the kinematic pressure, u and v are the velocity components in the x and y directions respectively, and $\nu = \dfrac{\mu}{\rho}$ is the kinematic viscosity; finally, we assume a constant density $\rho = 1$.
By employing a first order discretization in time, we obtain:

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} = -ADV_i^n - \nabla P_i^{n+1} + DIFF_i^n \quad \text{for } i = 1, 2 \tag{2}$$

where ADV_i^n is the advection term, ∇P_i^{n+1} is the pressure gradient term and $DIFF_i^n$ is the diffusion term.
The boundary conditions for the velocity field are the no-slip boundary conditions, except for the lid-driven boundary condition on the top wall:

$$u(x,0) = 0, \quad u(x,1) = 1, \quad u(0,y) = u(1,y) = 0 \tag{3}$$

$$v(x,0) = v(x,1) = 0, \quad v(0,y) = v(1,y) = 0$$

We then impose the homogeneus Neumann boundary conditions for the pressure field.

2.1 S-MAC method

We developed a code to solve the two dimensional incompressible Navier-Stokes equations using the simplified marker and cell (S-MAC) method, which is a finite difference-based method. This numerical method divides the problem into two stages, introducing the intermediate variables u_i^*.

2.1.1 Predictor step

The first stage is the predictor step, during which the intermediate velocity field (u^*, v^*) is computed. To do so, the code solves the following advection-diffusion equation, using a forward in time and centered in space explicit scheme:

$$\frac{u_i^* - u_i^n}{\Delta t} = -ADV_i^n + DIFF_i^n \quad \text{for } i = 1,2 \tag{4}$$

2.1.2 Corrector step

The second stage of the numerical method is the corrector step, where the velocity field at the new time step (u^{n+1}, v^{n+1}) is computed:

$$\frac{u_i^{n+1} - u_i^*}{\Delta t} = -\nabla P_i^{n+1} \quad \text{for } i = 1,2 \tag{5}$$

To obtain the pressure field P^{n+1}, the code solves a Poisson equation; in fact, by applying the divergence operator to the last equation (5) and using the continuity equation $(\nabla u_i^{n+1} = 0)$, we get:

$$\Delta P_i^{n+1} = \frac{\nabla u_i^*}{\Delta t} \tag{6}$$

The divergence of the intermediate velocity field is computed using a centered in space scheme. Considering the discrete Laplace operator, the Poisson equation (6) becomes a linear system, which the implemented code solves using the Jacobi iterative method, thus obtaining the pressure field P^{n+1}. Finally, the velocity field at the next time step (u^{n+1}, v^{n+1}) is computed using the following relation:

$$u_i^{n+1} = u_i^* - \Delta t \frac{\partial P^{n+1}}{\partial x_i} \quad \text{for } i = 1,2 \tag{7}$$

In particular, also in this case a centered in space scheme is employed to compute the gradient of pressure.

2.1.3 Implementation details

The code implements a square domain with side length $L = 1$ and a uniform 50×50 grid to discretize the domain. Specifically, a simple colocated grid is applied, with the velocity components and the pressure stored at the same grid points.
Key parameters include the time step $\Delta t = 10^{-4}$ s, the total number of iterations $N_t = 60000$, the kinematic viscosity $\nu = 0.01$ m^2/s and the Reynolds number $Re = \dfrac{L \cdot U}{\nu} = 100$.
After each time step, the implemented code calls a function to verify that the stability conditions are met. These conditions are crucial for ensuring the numerical stability and the accuracy of the simulation and are defined as follows:

$$\frac{\Delta t}{\Delta x^2 \cdot Re} \leq 0.25 \qquad \frac{\Delta t}{\Delta y^2 \cdot Re} \leq 0.25 \qquad \Delta t \left(\frac{|u|}{\Delta x} + \frac{|v|}{\Delta y} \right) \leq 1 \tag{8}$$

where the last one represents the Courant number.
The code was written in both `Python` and `MATLAB`; both versions are available on the author's *GitHub* repository [2].

2.1.4 Results

The results obtained from the simulation run with the custom code we developed are reported in this section. Figure (1) depicts the contour plot of the u velocity component in the simulated lid-driven cavity flow. The highest velocities are observed along the top boundary, corresponding to the moving lid; additionally, the effects of viscosity and diffusion are evident in the gradual decrease of the u velocity component away from the top of the cavity. A notable feature is the region of negative velocity towards the center, which indicates flow in the opposite direction of the lid movement and therefore the formation of a large primary vortex. Finally, the contour plot shows slight variations in the velocity near the bottom corners of the cavity, which could suggest the presence of two weaker secondary vortices.

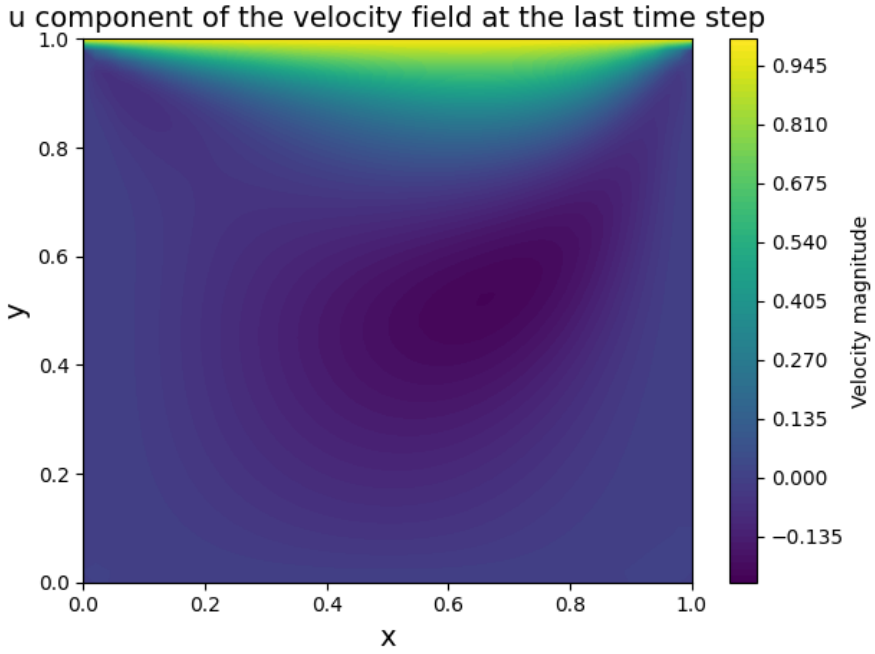

Figure 1: contour plot of the u velocity component in the simulated lid-driven cavity flow

The contour plots of the v velocity component and the pressure are depicted in figures (2) and (3) respectively.

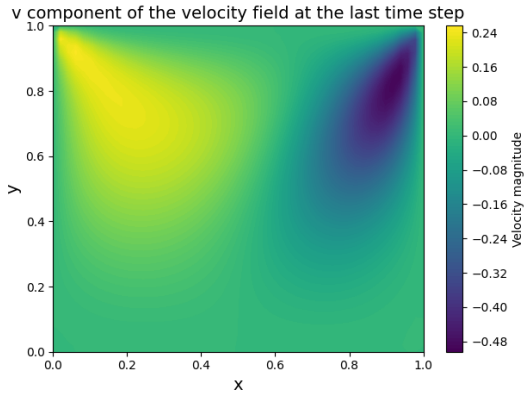

Figure 2: contour plot of the v velocity component in the simulated lid-driven cavity flow

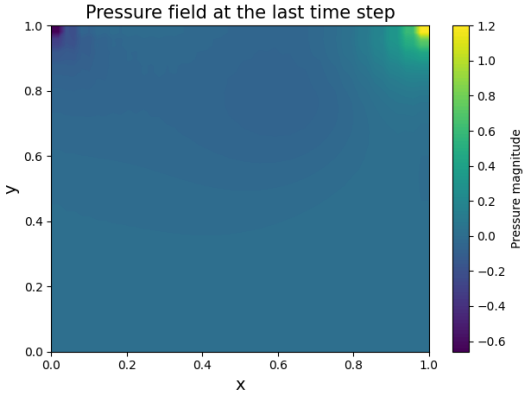

Figure 3: contour plot of the pressure field P in the simulated lid-driven cavity flow

In order to assess the accuracy and the reliability of the numerical simulation, its results were compared with those of *Ghia, Ghia and Shin* [1], obtained with $Re = 100$ on a 129×129 grid. As shown in figures (4) and (5), it can be observed that the simulation data closely matches the benchmark data; this strong agreement validates the results of the numerical simulation conducted using the custom implemented code, demonstrating both the precision and the robustness of the S-MAC method we employed.

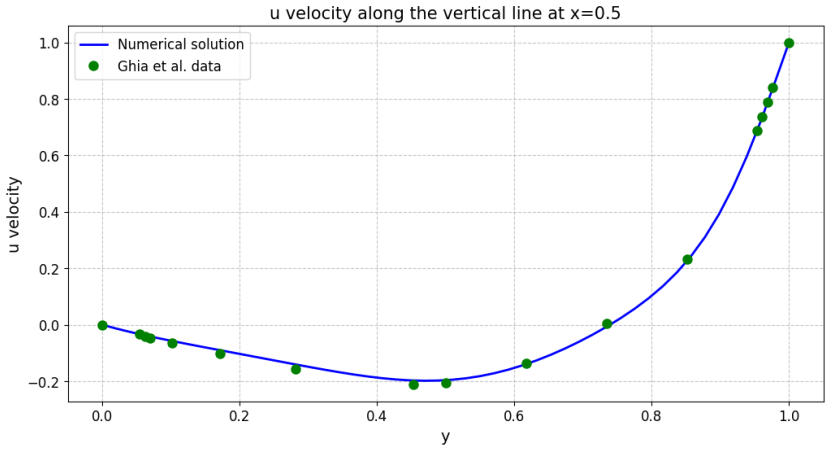

Figure 4: comparison of the results for the u velocity component along the vertical line $x = 0.5$ between the simulation data and the reference data from *Ghia, Ghia and Shin*

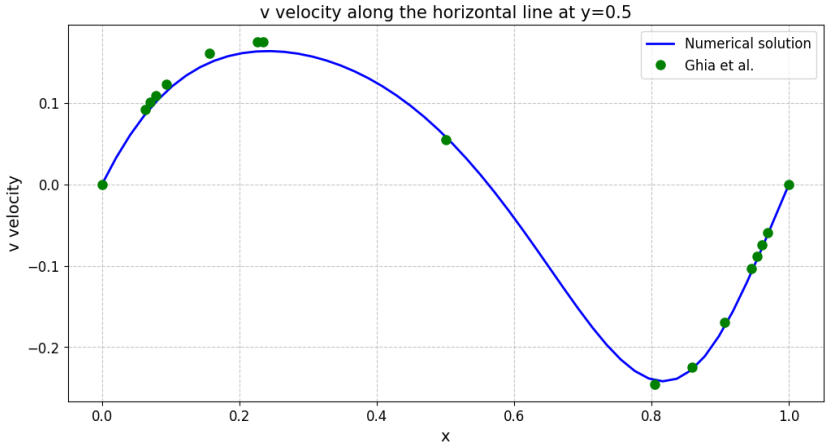

Figure 5: comparison of the results for the v velocity component along the horizontal line $y = 0.5$ between the simulation data and the reference data from *Ghia, Ghia and Shin*

2.2 OpenFoam

The lid-driven cavity flow was simulated using *OpenFoam* as well. The computational domain consisted of a two-dimensional square region with a side length $L = 1\ m$. The mesh was composed of four different blocks, each discretized into a 129×129 grid; different grading factors were employed in order to refine the grid near the corners and the boundaries of the cavity, providing higher resolution to capture the steep gradients in the flow field more accurately.
The simulation was conducted using the `icoFoam` solver; key parameters included the kinematic viscosity $\nu = 0.01\ m^2/s$, the Reynolds number $Re = 100$, the total simulation time $T = 6\ s$ and the time step $\Delta t = 10^{-3}\ s$.
All the *OpenFoam* scripts and configurations employed for this simulation are available on the author's *GitHub* repository [2].

2.2.1 Results

Figure (6) depicts the contour plot of the horizontal velocity component in the simulated lid-driven cavity flow. A comparable trend with respect to the plot in figure (1) can be immediately noticed. The red region at the top represents the high velocity of the fluid induced by the moving lid, while the dark blue areas indicate regions of low or negative velocity. The contour plot also reveals the effects of diffusion.
Additionally, the streamlines confirm the formation of primary and secondary vortices, consistently with the expected flow patterns for the lid-driven cavity flow. In fact, three distinct vortices can be noticed: the biggest one, called primary vortex, located above the center of the cavity, and two smaller ones, the secondary vortices, visible in the bottom left and bottom right corners.
Overall, the plot highlights the significant velocity gradients near the top moving lid and the formation of distinct vortical structures, characteristic of the lid-driven cavity flow.

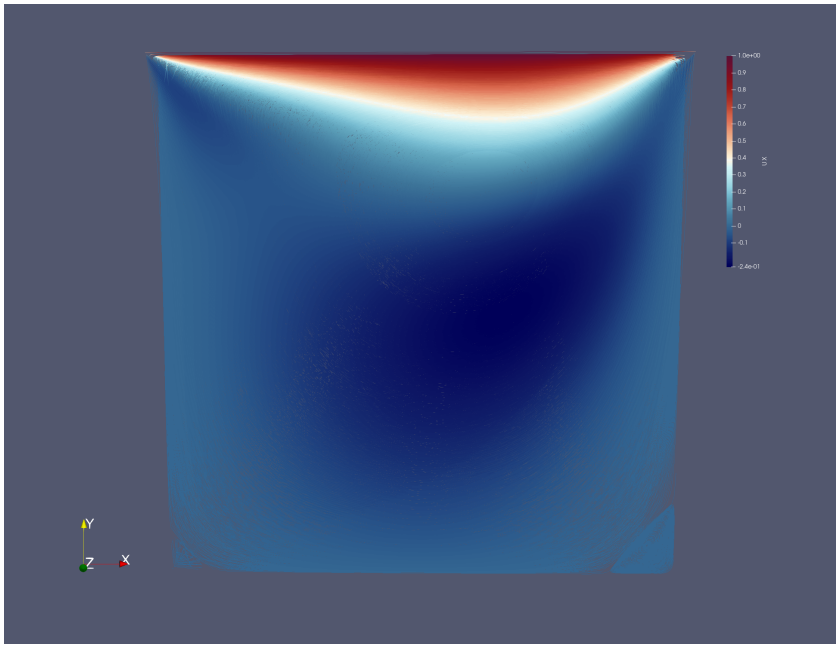

Figure 6: contour plot of the horizontal velocity component in the lid-driven cavity flow simulated using *OpenFoam*

Also in this case, the results obtained from the numerical simulation were compared with those of *Ghia, Ghia and Shin* [1], obtained with $Re = 100$ on a 129×129 grid.

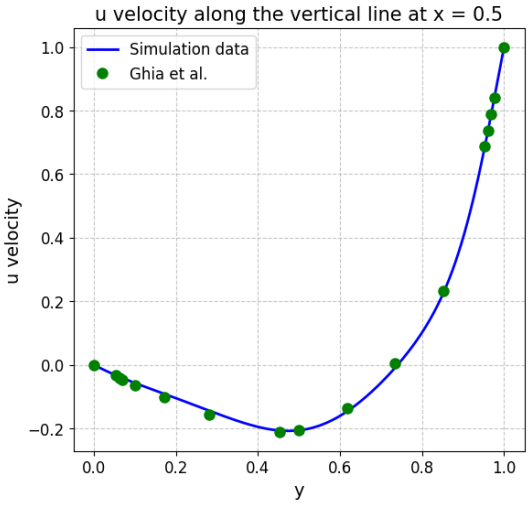

Figure 7: comparison of the results for the u velocity component along the vertical line $x = 0.5$ between the simulation data using *OpenFoam* and the reference data from *Ghia, Ghia and Shin*

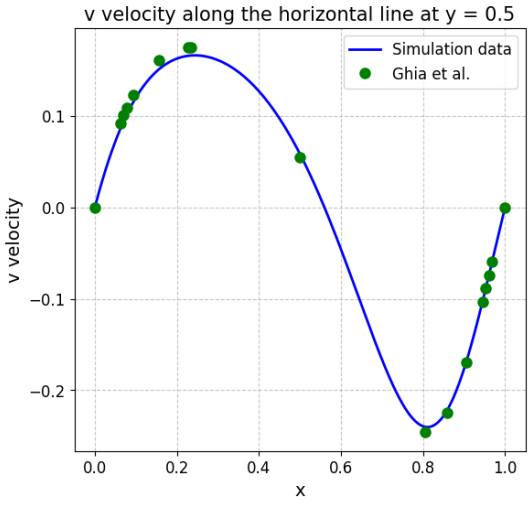

Figure 8: comparison of the results for the v velocity component along the horizontal line $y = 0.5$ between the simulation data using *OpenFoam* and the reference data from *Ghia, Ghia and Shin*

By looking at figures (7) and (8), it can be noticed that the simulated data shows a good agreement with the benchmark data, highlighting the accuracy of the numerical simulation.

As in the S-MAC method case, the comparison indicates that the simulation effectively captures the key features of the flow, including the peak velocities and the locations of flow reversal.

3 Poiseuille flow

The Poiseuille flow describes the planar flow of an incompressible fluid between two parallel plates separated by a distance h; it is characterized by a parabolic velocity profile, where the fluid velocity is highest at the center and zero at the plates due to the no-slip boundary conditions.

3.1 Analytical solution

Consider the two-dimensional incompressible Navier-Stokes equations:

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0 \tag{9}$$

$$\frac{\partial u}{\partial t} = -u\frac{\partial u}{\partial x} - v\frac{\partial u}{\partial y} - \frac{\partial P}{\partial x} + \nu\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\right) \tag{10}$$

$$\frac{\partial v}{\partial t} = -u\frac{\partial v}{\partial x} - v\frac{\partial v}{\partial y} - \frac{\partial P}{\partial y} + \nu\left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2}\right) \tag{11}$$

Since there is no vertical motion ($v = 0$), equation (11) simply becomes:

$$\frac{\partial P}{\partial y} = 0 \tag{12}$$

This indicates that the pressure is constant along the vertical direction. Moreover, considering the continuity equation (9), we obtain:

$$\frac{\partial u}{\partial x} = 0 \tag{13}$$

As a result, the velocity varies only along the y direction, so we write $u = u(y)$.
As the flow is steady and fully developed, we have:

$$\frac{\partial u}{\partial t} = 0 \qquad \frac{\partial u}{\partial x} = \frac{\partial^2 u}{\partial^2 x} = 0 \tag{14}$$

Applying these considerations to equation (10), we get:

$$-\frac{\partial P}{\partial x} + \nu\frac{\partial^2 u}{\partial^2 y} = 0 \tag{15}$$

Integrating this equation twice with respect to y returns:

$$\nu u - \frac{y^2}{2}\frac{\partial P}{\partial x} + Cy + D = 0 \tag{16}$$

where C and D are two constants. Applying the no-slip boundary conditions $u(0) = u(h) = 0$, we finally obtain the analytical solution:

$$u = -\frac{y}{\nu}\frac{\partial P}{\partial x}\left(\frac{h}{2} - \frac{y}{2}\right) \tag{17}$$

which indeed represents a parabolic profile.

3.2 Results

To simulate the Poiseuille flow, we used the computational fluid dynamics software *OpenFoam*. The computational domain consisted of a two-dimensional rectangular region of length $L = 4\ m$ and height $h = 1\ m$, representing the space between the two parallel plates. The mesh was uniformly divided into $N_x = 320$ cells along the length and $N_y = 80$ cells along the height, ensuring a fine resolution for capturing the flow characteristics.
The `cyclic` boundary condition was applied to simulate a fully developed flow, making the inlet and outlet patches periodic; additionally, the no-slip boundary condition was imposed to the top and bottom walls (`fixedWalls`), ensuring zero velocity at these walls.
The simulation was conducted using the `pimpleFoam` solver; key parameters included the kinematic viscosity $\nu = 0.01\ m^2/s$ and the average velocity $\bar{U} = 1\ m/s$. The total simulation time was set to $T = 6\ s$, with a time step $\Delta t = 10^{-3}\ s$.
All the *OpenFoam* scripts and configurations employed for this simulation are available on the author's *GitHub* repository [2].
Figure (9) shows the contour plot of the velocity magnitude in the simulated Poiseuille flow:

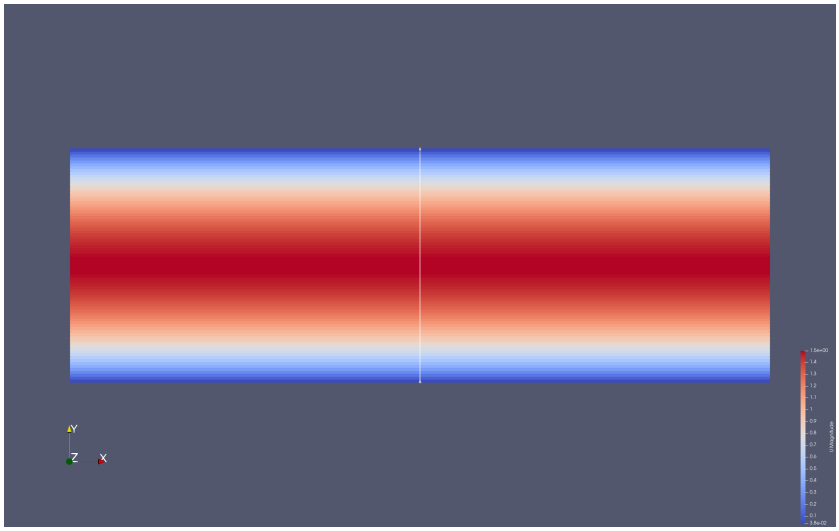

Figure 9: contour plot of the velocity magnitude in the simulated Poiseuille flow

By looking at the plot in figure (9), it can be seen that, as expected, the flow exhibits a parabolic velocity profile, with the highest velocity at the center of the pipe (shown in red) and decreasing gradually towards the walls (shown in blue), where the velocity is zero due to the no-slip boundary conditions. The symmetric nature of the flow is also evident.
The results we obtained were validated against the analytical solution (17) to ensure the accuracy of the simulation. The pressure gradient was computed using the following relationship, which makes use of the average velocity $\bar{U}$:

$$\frac{\partial P}{\partial x} = -\frac{12 \cdot \bar{U} \cdot \nu}{h^2} \tag{18}$$

Figure (10) presents a comparison between the simulation data and the analytical solution:

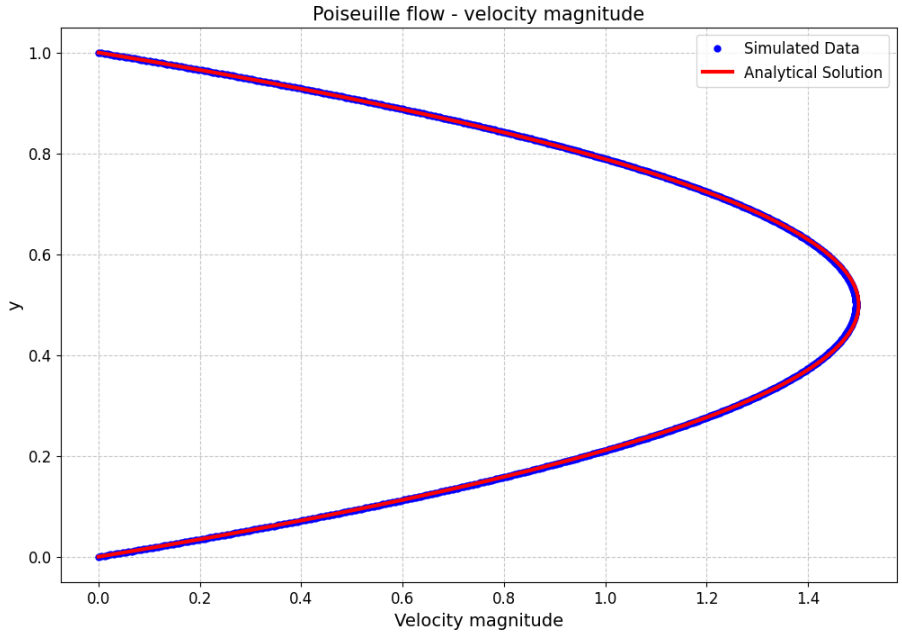

Figure 10: comparison between the simulation results and the analytical solution of the Poiseuille flow

Again, it can be noticed that the velocity profile is parabolic, as expected for a 2D Poiseuille flow: it is symmetric with respect to the center of the channel and the maximum velocity is located at its center. Moreover, we can observe that the analytical solution and the simulation data overlap almost perfectly, which means that the simulation results are accurate.
Overall, the plot in figure (10) demonstrates the alignment between the computational results and the theoretical expectations, confirming the reliability and the accuracy of the numerical simulation.

4 References

1. Ghia, Ghia, and Shin. *"High-Re solutions for incompressible flow using the Navier-Stokes equations and a multigrid method."* Journal of computational physics 48.3 (1982): 387-411

2. Piero Zappi, `Computational_Fluid_Dynamics_FinalProject` GitHub repository. URL: `https://github.com/PieroZ01/Computational_Fluid_Dynamics_FinalProject`