

Computational Fluid Dynamics via Semi-Implicit Method for Pressure-Linked Equations (SIMPLE)

This document is primarily based on a book by Patankar [3] and Versteeg et al. [4].

1 Lid Driven Cavity

Lid Driven Cavity (LDC) is a classic benchmark for incompressible flow solvers [2]. The most basic case is 2D square domain as illustrated in figure 1a. The test naturally extends to 3D where 5 out of 6 faces of the cube are no slip walls and the top face of the cube is the lid. The state at $t = 0$ is initialized in such a way that the cavity pressure in the bottom left corner is 0 and velocities within the inner cells are 0. The only driving force is through the lid. The goal of the simulation is to determine intermediate and final velocities.

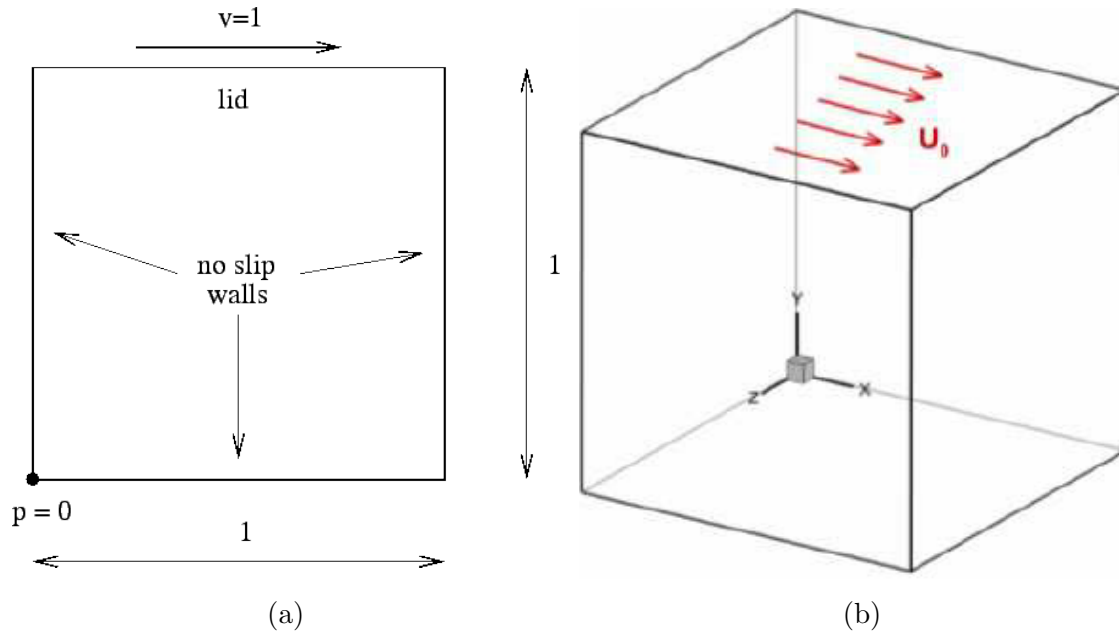


Figure 1: Lid Driven Cavity in 2D (a) and 3D (b)

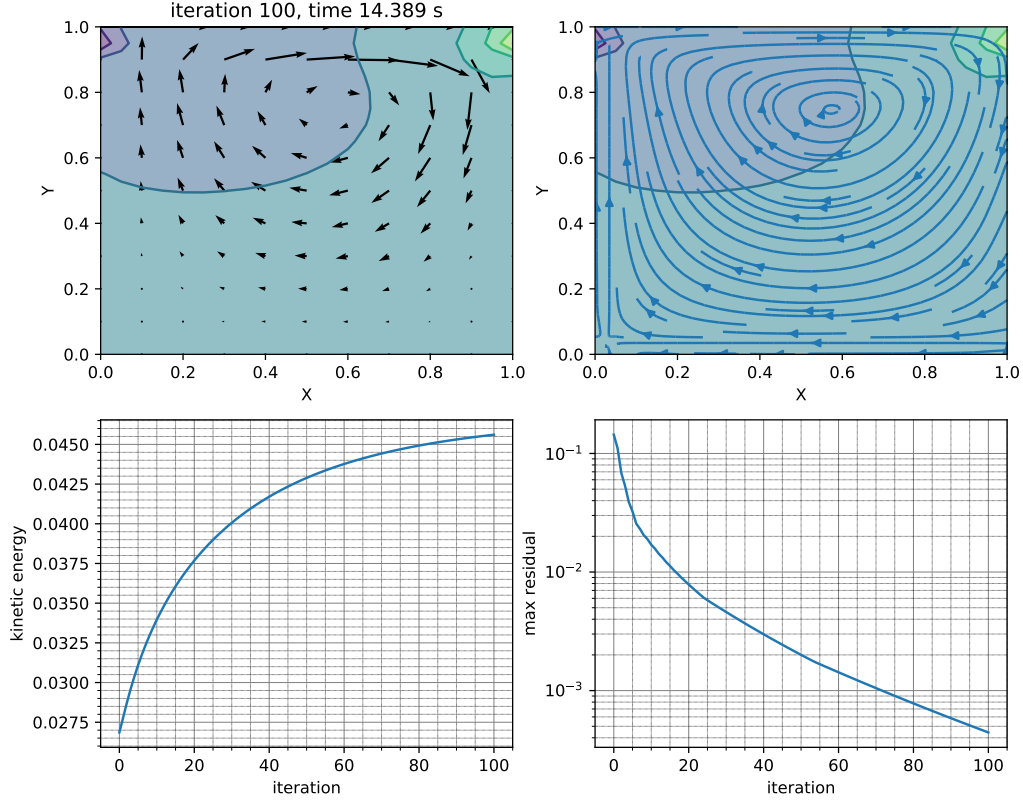


Figure 2: Lid Driven Cavity in 2D - simulation until convergence

Fig. 2 shows an example output of a small 2D simulation on a 20×20 grid. The top-left sub-figure shows the velocity vector field and the top-right one presents corresponding streamlines) at the end of simulation (100 time steps, roughly 14 seconds of CPU time). The pressure/density scalar field is described by color. The bottom-left plot shows how the total kinetic energy proportional to the velocities evolves in time until it reaches a steady state. Finally, the steady state can be identified by monitoring the maximum change in velocities in consecutive time steps (residual). The 3D simulation is analogous.

Parameters affecting the simulation:

- Grid size $X \times Y \times Z$
- Tolerance and/or maximum time steps
- Density ρ
- Dynamic Viscosity μ
- Velocity under-relaxation α_U
- Pressure under-relaxation α_p
- Lid Velocity U_l

- Linear solver (BiCGSTAB or other) and its internal parameters

i= 30, residual=0.003600, divergence=1.633301, time elapsed = 33.074540 s

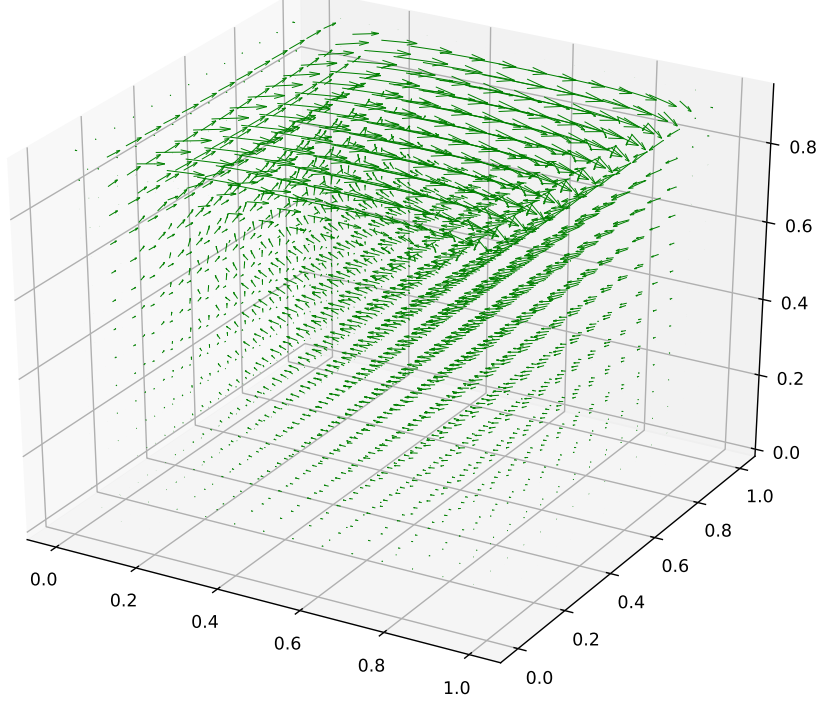


Figure 3: Lid Driven Cavity in 3D - 15 x 15 x 15 grid

2 Methodology

2.1 Navier-Stokes

The Navier-Stokes equations describe the motion of viscous fluids. For a single-phase flow with a constant density and viscosity they are:

$$\frac{\partial \mathbf{u}}{\partial t} = -(\mathbf{u} \cdot \nabla) \mathbf{u} - \frac{1}{\rho} \nabla p + \nu \nabla^2 \mathbf{u} + \mathbf{F} \quad (2.1)$$

Momentum Equation

$$\nabla \cdot \mathbf{u} = 0 \quad (2.2)$$

Continuity Equation

In Cartesian coordinate system, the solid body velocity \mathbf{u} can be resolved into components in the x , y and z directions. $\mathbf{F} = (f_x, f_y, f_z)$ represents any external force, ν is the kinematic viscosity of the fluid and ρ is its density.

$$\mathbf{u} = (u_x, u_y, u_z)$$

In order to simulate the motion of the fluid, we need to solve the momentum equation, which expresses the change of momentum with time. The continuity equation imposes the conservation of mass constraint. p is kinematic pressure $= P/\rho$, therefore have 4 equations and 4 unknowns (u_x, u_y, u_z, p) .

3 Finite Volume Method

Finite Volume Method (FVM) is one of the most popular method for solving the Navier-Stokes equations numerically. Finite volume methods extend on finite difference methods by imposing volume conservation. The computation domain is divided into many finite cells; conservation is imposed since the flux leaving each face of the cell must equal to the flux entering.

The region in which computations are to be performed is divided into a set of small cells having edge lengths δx , δy and δz . With respect to this set of computational cells, velocity components are located at the centre of the cell faces to which they are normal and pressure and temperature are defined at the centre of the cells. Cells are labeled with an index (i,j,k) which denotes the cell number as counted from the origin in the x , y and z directions respectively. Also $p(i,j,k)$ is the pressure at the centre of the cell (i,j,k) , while $u(i,j,k)$ is the x -direction velocity at the centre of the face between cells (i,j,k) and $(i+1,j,k)$ and so on [1].

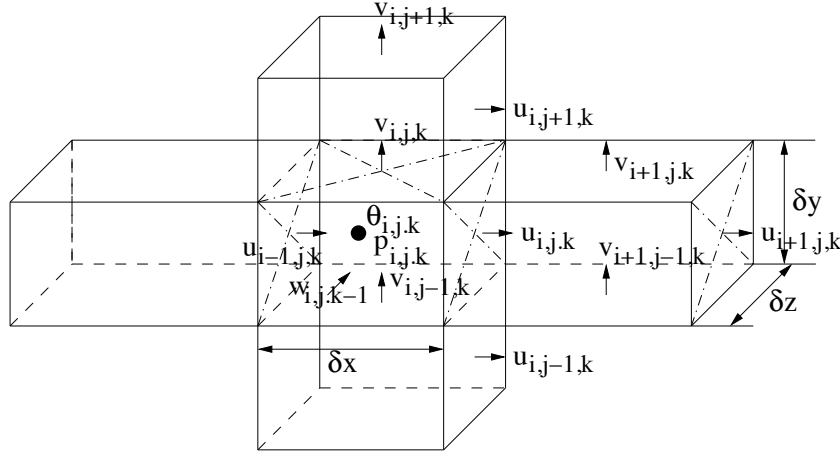


Figure 4: Three-dimentional staggered grid showing the locations of the discretized variables

4 SIMPLE algorithm

The SIMPLE (Semi-Implicit Method for Pressure-Linked Equations) allows to couple the Navier-Stokes equations with an iterative procedure.

Algorithm 1 SIMPLE algorithm[3]

```
1: Guess the initial pressure field  $p^*$ 
2: while not converged do
3:   Set the boundary conditions
4:   Solve the discretized momentum equations to obtain  $u^*, v^*$  and  $w^*$ 
5:   Solve the  $p'$  equation (BiCGStab or similar)
6:   Calculate  $p = p' + p^*$ 
7:   Calculate  $u, v, w$  from  $u^*, v^*, w^*$  using velocity-correction formulas
8:   Update the boundary conditions
9:    $p^* = p$ 
10: end while
```

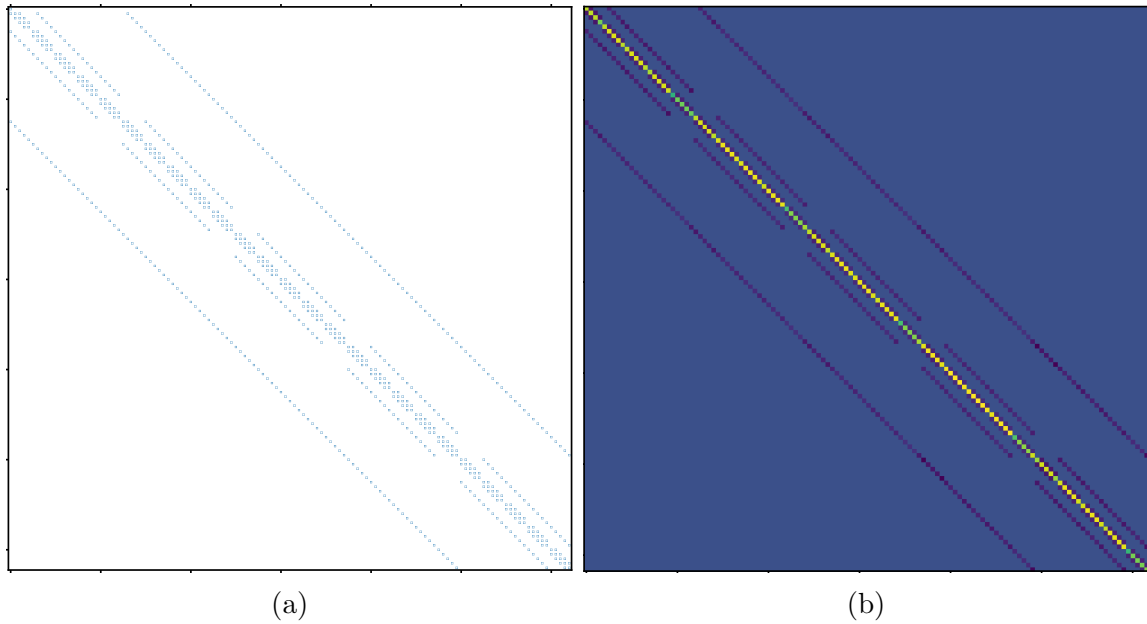


Figure 5: Solving the pressure equation (line 5) is the main computational task. (a) shows the sparsity pattern of the matrix involved, while (b) shows the values; an example for the 5x5x5 case

5 References

- [1] Chapter 6 solution of navier-stokes equations for incompressible flows using simple and mac algorithms 6 . 1. 2015.
- [2] U Ghia, K.N Ghia, and C.T Shin. High-re solutions for incompressible flow using the navier-stokes equations and a multigrid method. *Journal of Computational Physics*, 48(3): 387 – 411, 1982. ISSN 0021-9991. doi: [https://doi.org/10.1016/0021-9991\(82\)90058-4](https://doi.org/10.1016/0021-9991(82)90058-4). URL <http://www.sciencedirect.com/science/article/pii/0021999182900584>.
- [3] Suhas V Patankar. *Numerical heat transfer and fluid flow*. Series on Computational Methods in Mechanics and Thermal Science. Hemisphere Publishing Corporation (CRC Press, Taylor & Francis Group), 1980. ISBN 978-0891165224. URL <http://www.crcpress.com/product/isbn/9780891165224>.
- [4] Henk Kaarle Versteeg and Weeratunge Malalasekera. *An introduction to computational fluid dynamics - the finite volume method*. Addison-Wesley-Longman, 1995. ISBN 978-0-582-21884-0.