ME663 - Computational Fluid Dynamics
Assignment 1

Tommaso Bocchietti

A.Y. 2023/24 - W24

UNIVERSITY OF
WATERLOO

Contents

List of Figures

List of Tables

1 Requests

This assignment was about writing a computer program using the language we prefer, to solve the incompressible Navier-Stokes equations based on Vanka's Symmetric Coupled Gauss-Seidel (SCGS) method [2].
The objective here is to find out the numerical solutions for a flow in a square cavity with the top wall moving at a constant velocity U_{lid}. Different suggestions / requests were given, such as:

- Use UDS, Hybrid, and/or QUICK convection schemes on a mesh of $40x40$ and $80x80$ nodes at $Re = 400$ and $Re = 1000$, where $Re = \frac{U_{lid} \cdot L}{\nu}$ and L is the length of the cavity.
- Set the convergence criterion to 10^{-4}.
- Report the wall-clock time and total number of iterations for each case.
- Compare your results in terms of $u(y)@x = 0.5L$ and $v(x)@y = 0.5L$ with results in Tables I and II from *Ghia et al.* [1] paper.
- What is the optimal under-relaxation factor for each case?
- If you try more than one convection scheme, which scheme is more accurate compared to *Ghia et al.* [1] 'exact solutions'?
- Use $2^{nd} - order$ and $4^{th} - order$ schemes to approximate the diffusion term $\nu \left[\frac{\partial^2(\phi)}{\partial x^2} + \frac{\partial^2(\phi)}{\partial y^2} \right]$ in the Navier-Stokes equation.

2 Methods

For this assignment, we have chosen to implement the code using C-language, and to implement different schemes and methods to solve the incompressible Navier-Stokes equations. In particular, we have implemented the following schemes for the convection term:

- Upwind Differencing Scheme (UDS)
- Hybrid Scheme
- Quadratic Upstream Interpolation for Convective Kinematics (QUICK)

We have also implemented the following schemes for the diffusion term:

- Second-order central difference scheme
- Fourth-order central difference scheme

Instead, for the methods to solve the linear system of equations, we have gone a little further and implemented the following methods:

- Symmetric Coupled Gauss-Seidel (SCGS)
- Semi-Implicit Method for Pressure Linked Equations (SIMPLE)

The main idea behind the implementation of these schemes and methods was to compare the various combinations of those, and to find out which one would be the most accurate and efficient for the given problem.
In the following sections, we will describe the implementation of each of these schemes and methods, and we will also present the results obtained from the simulations.
Complete code can be found at `https://github.com/Bocchio01/CFD_Simulation_Engine`, along with its documentation and instructions on how to run it.

3 Derivation of discretized governing equations

In this section, we will derive the discretized governing equations for the incompressible Navier-Stokes equations, which will be used to solve the problem at hand.
The set of incompressible Navier-Stokes equations is given by:

$$\nabla \cdot \mathbf{u} = 0$$

$$\rho \frac{\partial \mathbf{u}}{\partial t} + \rho \mathbf{u} \cdot \nabla \mathbf{u} = -\nabla p + \mu \nabla^2 \mathbf{u} + \mathbf{f}$$

In the rest of the document, the following hypotheses will be considered:

- Steady-state problem: $\frac{\partial \mathbf{u}}{\partial t} = 0$
- Constant density: $\rho = \text{const}$
- Constant dynamic viscosity: $\mu = \text{const}$
- Zero body forces: $\mathbf{f} = 0$

Based on these hypotheses, the incompressible Navier-Stokes equations can be simplified and expanded as follows:

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0 \tag{1}$$

$$\frac{\partial uu}{\partial x} + \frac{\partial vu}{\partial y} = -\frac{\partial p}{\partial x} + \nu \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) \tag{2}$$

$$\frac{\partial uv}{\partial x} + \frac{\partial vv}{\partial y} = -\frac{\partial p}{\partial y} + \nu \left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} \right) \tag{3}$$

Where $\nu = \frac{\mu}{\rho}$ is the kinematic viscosity and $p = \frac{p}{\rho}$ is the non-dimensional pressure.
Obviously, to solve the problem using a discrete calculator, the equations must be therefore discretized.

3.1 Finite Volume Method

The Finite Volume Method (FVM) is a numerical technique used to discretize partial differential equations, and is particularly well suited for the discretization of the Navier-Stokes equations.
The idea here is to divide the domain into a set of control volumes, and then integrate the governing equations over each control volume. The resulting set of equations will be a set of algebraic equations, which can be solved using a discrete calculator.

3.1.1 Control volumes

Before proceeding with the discretization of the governing equations, we need to define what a control volume is and the notations used in the rest of the document.
In particular, we will assume from now on to have a Cartesian grid, with a uniform mesh spacing in both the x and y directions.
From the Figure 1, we can appreciate graphically how the domain is divided.

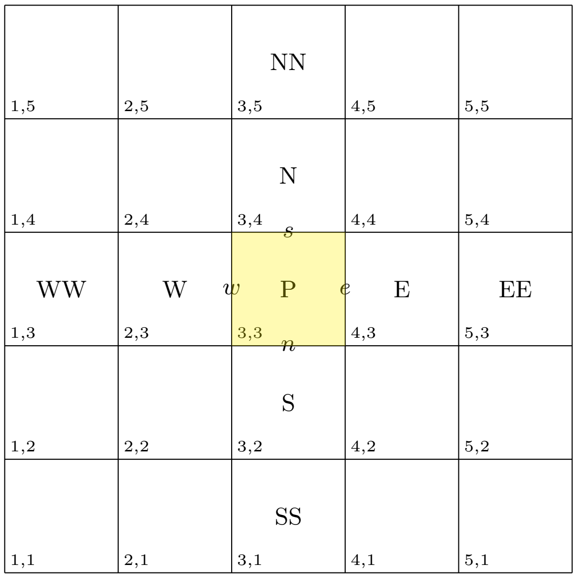

Figure 1: Control volumes and control volume faces.

In particular, Figure 1 shows:

- A grid of control volumes, with the subscript (i, j) indicating the position of the control volume in the x and y directions, respectively. For example, the control volume in the center of the grid has indices $(i, j) = (3, 3)$.
- The control volume centers with capital letters, $P, N, S, E, W, \ldots$.
- The control volume faces with lowercase letters, n, s, e, w.

Notice also that the capital letters always refers to a relative position with respect to the control volume in consideration. For example, P refers to the control volume in consideration, N refers to the control volume to the north of P, and so on. For this reason, in Figure 1, the control volume P is highlighted in yellow so to indicate that is the control volume in consideration.

3.1.2 Staggered grid and L_{shape}

As we will see later during the formulation of the solving solution, for the purpose of this work, we will use a specific type of grid, called staggered grid.
We can also give a brief definition of the two types of grids available in the literature, which are:

- **Collocated grid**: all the variables are located at the same point in the control volume (e.g. the center of the control volume).
- **Staggered grid**: the variables are located at different points in the control volume (e.g. the velocity components are located at the center of the faces of the control volume, and the pressure is located at the center of the control volume).

Given the formulation of the staggered grid, it's now useful to define the so called L_{shape}, which is a frame used to define in a compact and clear way the position of the variables in the control volume.
The L_{shape} has been reported for control volume P in Figure 2.

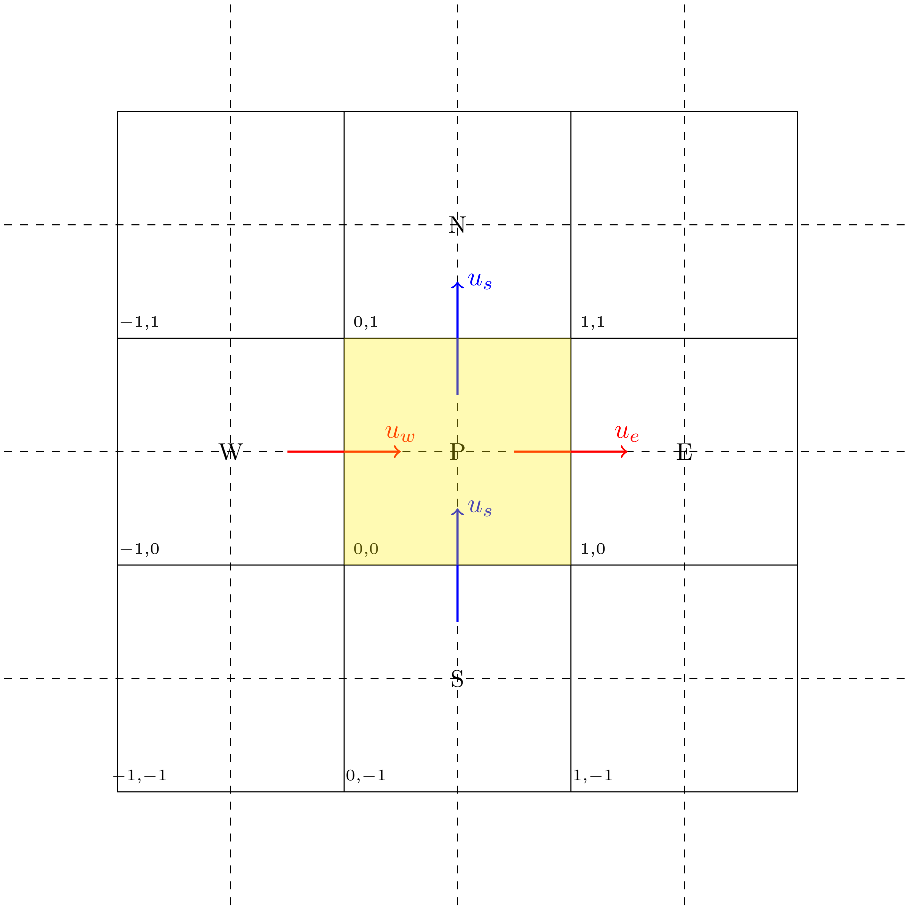

Figure 2: L_{shape} for control volume P.

Basically, the L_{shape} for control volume P links the velocity components to the control volume P itself, and it's used to define the indexes of the system.
In particular, the same Figure 2 can be represented using the index notations, as shown in Figure 3.

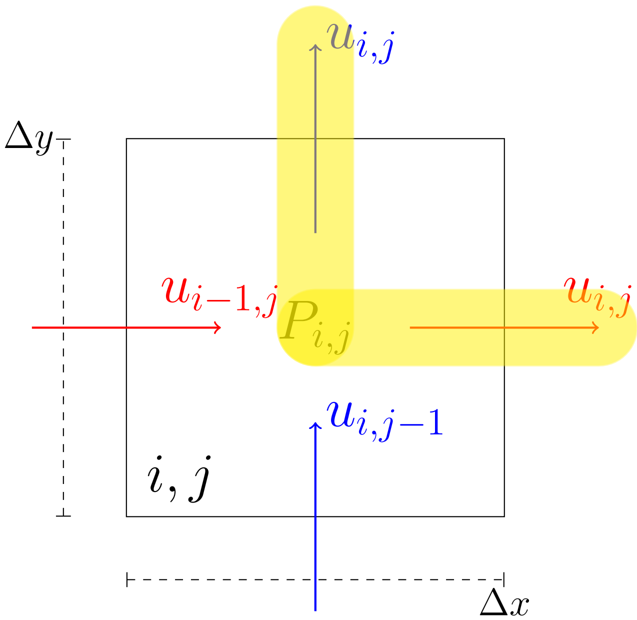

Figure 3: L_{shape} for control volume P using index notations.

From Figure 3, we can appreciate how the L_{shape} works well with index notations, and can be useful when working purely with indexes to refer to the variables.

3.2 Application of the Finite Volume Method

Having defined our working framework, we can now proceed to the application of the Finite Volume Method (FVM) to the incompressible Navier-Stokes equations 3.
To do so, we start by giving the general integral form of the governing equations, and then proceed to the discretization of the convection, diffusion and source terms separately.
In particular, assuming from now on to have a Cartesian grid, with a uniform mesh spacing in both the x and y directions, the (FVM) reads as follows:
From now on, we will focus on the x direction, given that the treatment for the y direction is analogous.
The first step is to integrate the governing equations over each control volume.
Our discretized set of equations will be:

$$\int_V u\frac{\partial u}{\partial x}dV + v\frac{\partial u}{\partial y}dV - \frac{1}{\rho}\frac{\partial p}{\partial x}dV - \nu\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\right)dV - f_x dV = 0 \tag{4}$$

$$\int_V u\frac{\partial v}{\partial x}dV + v\frac{\partial v}{\partial y}dV - \frac{1}{\rho}\frac{\partial p}{\partial y}dV - \nu\left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2}\right)dV - f_y dV = 0 \tag{5}$$

We can now proceed to the discretization of the convection, diffusion and source terms separately.
The rest of the treatment will be done considering just the x direction, given that the treatment for the y direction is analogous.
Our starter equation is:

$$\int_V \underbrace{u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y}}_{\text{Convection term}} - \underbrace{\frac{1}{\rho}\frac{\partial p}{\partial x}dV}_{\text{Pressure term}} \underbrace{-\nu\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\right)dV}_{\text{Diffusion term}} - \underbrace{f_x dV}_{\text{Source term}} = 0 \tag{6}$$

Before proceeding, we need to define the control volume and the control volume faces geometrically.

References

[1] Urmila Ghia, Karman Ghia, and C. T. Shin. High-re solutions for incompressible flow using the navier-stokes equations and a multigrid method. *Journal of Computational Physics*, 48:387–411, 1982.

[2] S.P Vanka. Block-implicit multigrid solution of navier-stokes equations in primitive variables. *Journal of Computational Physics*, 65(1):138–158, 1986.