Polytechnic University of Catalonia

The School of Industrial, Aerospace and Audiovisual Engineering of Terrassa

REPORT

Study for the computational resolution of the conservation equations of mass, momentum and energy. Application to different aeronautical and industrial engineering problems.

Thesis submitted in order to obtain the degree in Aerospace Technology Engineering

Author:
Francesc Hernández Garcia

Director:
Asensio Oliva Llena

Co-Director:
Jordi Vera i Fernández

Barcelona. Spring, 2022

Acknowledgements

Primerament, agrair als professors del CTTC tota l'ajuda, consells i temps que han dedicat a fer que aquest treball sigui possible. Especialment al meu director i codirector, els professors Assensi Oliva i Jordi Vera, per la seva incansable feina i experiència. També vull mencionar als professors Carles David Pérez Segarra, Francesc Xavier Trias i Joaquim Rigola per haver-me transmès part del seu coneixement, motivant-me a entrar al món de la mecànica de fluids. Als meus tutors durant la meva etapa al sincrotró ALBA, per transmetre'm la seva passió per la transferència de calor. Altrament, agrair a la meva família i amistats el seu constant suport durant el transcurs dels anys.

Francesc Hernández Garcia

Abstract

The following final degree dissertation composes a study of the computational resolution of the Navier-Stokes equations. The objective of the work is to introduce the student to the field of Computational Fluid Dynamic (CFD) simulations. This is done by creating self-made codes able to solve problems proposed by the Heat and Mass Transfer Technological Center (CTTC) of the Polytechnic University of Catalonia (UPC).

The equations of mass, momentum and energy are solved by means of the Finite Volume Method (FVM) with an algorithm based on the Fractional Step Method (FSM) for incompressible fluids. All the codes are self-programmed and verified by the student in C++ language. Emphasis is made in understanding the different theoretical and computational implications of the Navier-Stokes equations. The different problems are chosen specifically to treat as many aspects of the incompressible Navier-Stokes equations as possible. Their difficulty increases progressively, starting from a simple conduction heat transfer problem and ending in the study of turbulence.

The contribution of the convective and diffusive terms is deeply analyzed, firstly by solving a pure diffusion case and then by finding the numerical solution of a general convection-diffusion equation. The Fractional Step Method is employed to solve both internal flow (with forced and natural convection) and external flow (around a square cylinder). Different aspects about turbulence are studied and implemented to the resolution of the Burgers equation and a three-dimensional channel flow. To finalize the work, a proposal for future steps is given in relation to a more advanced research project, based on a deeper study of turbulence models and High Performance Computing (HPC).

Resum

El següent treball de fi de grau presenta un estudi sobre la resolució computacional de les equacions de Navier-Stokes. L'objectiu del treball és introduir a l'estudiant al camp de les simulacions de Dinàmica de Fluids Computacional (CFD). Això s'aconsegueix mitjançant codis programats pel mateix estudiant capaços de resoldre problemes proposats pel Centre Tecnològic de Transferència de Calor (CTTC) de la Universitat Politècnica de Catalunya (UPC).

Les equacions de massa, quantitat de moviment i energia es resolen mitjançant el Mètode de Volums Finits (FVM) amb un algoritme basat en el Mètode de Pas Fraccionari (FSM) per fluids incompressibles. Tots els codis són programats i verificats per l'estudiant en llenguatge de programació C++. Es fa èmfasis en la comprensió de les diferents implicacions teòriques i computacionals de les equacions de Navier-Stokes. Els diferents problemes s'escullen específicament per tractar tants conceptes de les equacions de Navier-Stokes en règim incompressible com sigui possible. La seva dificultat incrementa progressivament, començant amb un cas de transferència de calor per conducció simple i acabant en l'estudi de la turbulència.

La contribució dels termes convectius i difusius s'analitza exhaustivament, primerament resolent un cas de difusió pura i després trobant la solució numèrica d'una equació general de convecció-difusió. El Mètode de Pas Fraccionari s'utilitza per resoldre tant flux intern (amb convecció forçada i natural) com flux extern (al voltant d'un cilindre quadrat). S'estudien i implementen diferents aspectes sobre turbulència mitjançant la resolució de l'equació de Burgers i un problema tridimensional en canal turbulent. Per acabar el treball, es fa una proposta sobre futurs passos relacionats amb un projecte de recerca més avançat, basat en un estudi més profund sobre els models de turbulència i la Computació d'Altes Prestacions (HPC).

Contents

List of Figures

List of Tables

1. Introduction

1.1. Aim of the study

The aim of this project is to study the numerical resolution of the incompressible Navier-Stokes equations. In order to do so, the mathematical formulation of the equations of mass, momentum and energy is presented, analyzed and applied to several problems proposed by the Heat and Mass Transfer Technological Center (CTTC) [1]. The complexity of the mentioned problems increases progressively, starting from a simple pure diffusion case, passing through the resolution of a laminar flow and ending in a three-dimensional turbulent case. These problems are solved by means of codes self-developed in C++ language, and the obtained data is post-processed with MATLAB.

1.2. Scope of the study

This project includes:

- Study and deduction of the Navier-Stokes equations, taking as starting point the Reynolds transport theorem. Understanding of the nomenclature and vectorial notation.
- Introduction to the theoretical aspects of the Finite Volume Method (FVM). Discretization of the convective and diffusive terms of a generic convection-diffusion equation.
- Numerical resolution of a two-dimensional pure diffusion equation for a heat transfer conduction problem with different materials.
- Resolution of a generic convection-diffusion equation. Study of different convective interpolation schemes found on the literature. Application to the Smith-Hutton problem.
- Study of the theoretical aspects of the Fractional Step Method (FSM). Understanding of the mathematical formulation and discretization of the Navier-Stokes equations.
- Application of the Fractional Step Method to the Lid Driven Cavity problem for different Reynolds numbers and verification of the algorithm.
- Resolution of a natural convection case by considering a Boussinesq fluid in the Differentially Heated Cavity problem. Simultaneous resolution of a convection-diffusion equation for the temperature field and the Fractional Step Method for the pressure and velocity fields.
- Application of the Fractional Step Method to an external flow around a two-dimensional square cylinder for low Reynolds numbers.
- Introduction to turbulence. Understanding of the Kolmogorov hypothesis and the concept of energy cascade. Computational resolution of the Burgers equation in a one-dimensional domain.
- Numerical resolution of a turbulent three-dimensional Channel Flow problem by employing a LES Smagorinsky model.
- Verification of all the programmed codes by comparing the results with reference data.
- Study of the environmental impact of the present work.

Otherwise, this project does not include:

- Study of the compressibility effects of the fluid.
- Implementation of any other methodology other than the Finite Volume Method.
- Experimental frame in the laboratory.

1.3. Requirements

The requirements for this project are summarized below.

- All the codes must be programmed by the student in C++ language. Nonetheless, the post-processing of the results can be performed utilizing other software.
- The developed code must be compilable and executable without the definition of any user input.
- The results of the simulations will be verified by comparing them with reference data extracted from the literature or given by the CTTC, in accordance with the intellectual property laws.

1.4. Justification

The computational engineering methods are in high demand these days. Not only for the necessity of diminishing the elevated cost of experimental analysis, but also for the need of validating complex physical problems with the computer. In this scenario, Computational Fluid Dynamics (CFD) appeared in the latter second half of the twentieth century with the aim of predicting the behaviour of a fluid in motion [2]. In order to do so, the equations of Navier-Stokes are solved computationally by means of numerical methods. These equations are expressed in differential form, as no exact solution has been found yet for their generic formulation.

The problems of the first algorithms at that time were of main concern. On the one hand, the computational effort for the resolution of these differential equations was extremely high for the contemporaneous hardware. Dealing with irregular boundary conditions was also a complex problem, which led to the development of specialized pre-processing software. Solution convergence was slow and numerical diffusion appeared due to the difficulty of meshing complex geometries. Furthermore, the treatment of the time-dependent term of the Navier-Stokes equations was still rudimentary and inefficient.

Aiming to create sustainable, easy-to-use and efficient software able to run on the most basic personal computers, an entire generation of engineers focused their work on solving the aforementioned problems. Thanks to this, in the present time there is a diverse spectrum of specialized software able to run CFD simulations for a large amount of aeronautical, mechanical, civil and medical applications (among others). Despite this, continuing to generate knowledge on the numerical resolution of differential equations such as the Navier-Stokes is of main concern for the engineering community.

The present final degree dissertation is framed under this context. With the help and guidance of CTTC (which leads the Spanish research on this working field), the student is introduced to CFD development, creating self-made codes and analyzing the results.

2. Mathematical Formulation

In the present section, the Navier-Stokes equations of mass conservation, momentum and energy are introduced in their integral and differential notation. The hypotheses made in order to obtain the equations to be solved in the present dissertation are presented. Most of the information is extracted from reference [3].

2.1. Mass conservation equation

The equation of mass conservation in its integral form, in an open system for a control volume in motion with velocity $\mathbf{u_b}$, is defined in equation (1). The notation CV and CS stands for control volume and control surface, respectively.

$$\frac{\partial}{\partial t}\int_{CV}\rho\, dV + \oint_{CS}\rho\,(\mathbf{u}-\mathbf{u_b})\cdot\mathbf{n}\, dS = 0 \tag{1}$$

The first term refers to the change in time of the mass contained by the control volume, while the second term stands for the exchange of mass through its surface due to the relative velocity of the fluid with the control volume. Performing an analysis over a differential control volume in Cartesian's coordinates, for a static control volume ($\mathbf{u_b} = 0$), the differential form is obtained[1].

$$\frac{\partial \rho}{\partial t} + \frac{\partial\,(\rho u)}{\partial x} + \frac{\partial\,(\rho v)}{\partial y} + \frac{\partial\,(\rho w)}{\partial z} = 0 \tag{2}$$

If the flow is incompressible ($\frac{\partial \rho}{\partial t} = 0$), equation (2) yields equation (3).

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} = 0 \tag{3}$$

In compacted vectorial notation, the divergence of the velocity is zero. The condition of incompressibility turns the velocity field into a divergence-free space by means of the mass conservation equation.

$$\nabla\cdot\mathbf{u} = 0 \tag{4}$$

2.2. Linear momentum equations

The linear momentum equation in its integral form, for an inertial system of coordinates, in a control volume in motion for an open system is defined in equation (5).

$$\frac{\partial}{\partial t}\int_{CV}\mathbf{u}\rho\, dV + \oint_{CS}\mathbf{u}\rho\,(\mathbf{u}-\mathbf{u_b})\cdot\mathbf{n}\, dS = \oint_{CS}\mathbf{f_{(n)}}\, dS + \int_{CV}\mathbf{f_b}\, dV \tag{5}$$

[1] Note that the velocity vector in three dimensions is defined as $\mathbf{u} = \begin{bmatrix} u \\ v \\ w \end{bmatrix}$.

The first term of the left hand side (LHS) stands for the change of momentum in time, while the second term stands for the exchange of momentum through the control surface of the control volume. The right hand side (RHS) refers to the sum of surface forces ($\mathbf{f_{(n)}}$) and body forces ($\mathbf{f_b}$) over the control surface and control volume, respectively. This equation is analogous to the Newton's second law for the fluid motion.

In regard to the surface forces, these are the pressure and tangential stresses[2], defined as $\mathbf{f_{(n)}} = -p\mathbf{n}+\mathbf{n}\cdot\boldsymbol{\tau}$. The body forces are mainly caused by the gravity and electromagnetic forces[3] $\mathbf{f_b} = \rho\mathbf{g}+\mathbf{f^e}$. Taking this into account for a static control volume, equation (6) is obtained.

$$\frac{\partial}{\partial t}\int_{CV}\mathbf{u}\rho\,dV + \oint_{CS}\mathbf{u}\rho\mathbf{u}\cdot\mathbf{n}\,dS = -\oint_{CS}p\mathbf{n}\,dS + \oint_{CS}\mathbf{n}\cdot\boldsymbol{\tau}\,dS + \int_{CV}\rho\mathbf{g}\,dV + \int_{CV}\mathbf{f^e}\,dV \tag{6}$$

Performing an analysis over a differential control volume in Cartesian's coordinates, a set of equations in differential notation is obtained (one for each dimension).

$$\frac{\partial\rho u}{\partial t} + \frac{\partial(\rho uu)}{\partial x} + \frac{\partial(\rho vu)}{\partial y} + \frac{\partial(\rho wu)}{\partial z} = -\frac{\partial p}{\partial x} + \frac{\partial\tau_{xx}}{\partial x} + \frac{\partial\tau_{yx}}{\partial y} + \frac{\partial\tau_{zx}}{\partial z} + \rho g_x + f_x^e \tag{7a}$$

$$\frac{\partial\rho v}{\partial t} + \frac{\partial(\rho uv)}{\partial x} + \frac{\partial(\rho vv)}{\partial y} + \frac{\partial(\rho wv)}{\partial z} = -\frac{\partial p}{\partial y} + \frac{\partial\tau_{xy}}{\partial x} + \frac{\partial\tau_{yy}}{\partial y} + \frac{\partial\tau_{zy}}{\partial z} + \rho g_y + f_y^e \tag{7b}$$

$$\frac{\partial\rho w}{\partial t} + \frac{\partial(\rho uw)}{\partial x} + \frac{\partial(\rho vw)}{\partial y} + \frac{\partial(\rho ww)}{\partial z} = -\frac{\partial p}{\partial z} + \frac{\partial\tau_{xz}}{\partial x} + \frac{\partial\tau_{yz}}{\partial y} + \frac{\partial\tau_{zz}}{\partial z} + \rho g_z + f_z^e \tag{7c}$$

This set of equations can be expressed in compacted vectorial notation in order to obtain equation (8).

$$\frac{\partial(\rho\mathbf{u})}{\partial t} + \nabla\cdot(\rho\mathbf{uu}) = -\nabla p + \nabla\cdot\boldsymbol{\tau} + \rho\mathbf{g} + \mathbf{f^e} \tag{8}$$

There are some important points to take into consideration:

- The thermophysical properties will be treated as constants.
- The electromagnetic forces are not going to be considered.
- As the fluid is incompressible, the density gets out of the partial derivatives of the LHS, yielding the material derivative of the velocity $\rho\frac{D\mathbf{u}}{Dt} = \rho\frac{\partial\mathbf{u}}{\partial t} + \rho\nabla\cdot(\mathbf{uu}) = \rho\frac{\partial\mathbf{u}}{\partial t} + \rho(\mathbf{u}\cdot\nabla)\mathbf{u}$.
- The gravity term can be treated by means of the Boussinesq hypothesis, further explained in Section 6.2.1. This introduces the concept of coefficient of volumetric expansion, as $\rho\mathbf{g} \approx -\rho\beta(T-T_\infty)\mathbf{g} + \nabla p_g$. The parameter T_∞ is a reference value for the temperature, assigned to the constant density ρ and the pressure p_g.

[2]Note that $\boldsymbol{\tau}$ is a symmetric square tensor of third order (in three-dimensions), defined as $\boldsymbol{\tau} = \begin{bmatrix}\tau_{xx} & \tau_{xy} & \tau_{xz}\\ \tau_{yx} & \tau_{yy} & \tau_{yz}\\ \tau_{zx} & \tau_{zy} & \tau_{zz}\end{bmatrix}$.

[3]In their most generic form, the gravity is defined as $\mathbf{g} = \begin{bmatrix}g_x\\ g_y\\ g_z\end{bmatrix}$ and the electromagnetic forces as $\mathbf{f^e} = \begin{bmatrix}f_x^e\\ f_y^e\\ f_z^e\end{bmatrix}$.

- The viscous forces term is treated by means of the Stoke's viscous law for Newtonian fluids, further explained in Section 7.2.2. This introduces the dynamic viscosity to the equation, as $\nabla \cdot \boldsymbol{\tau} \approx \mu \nabla^2 \mathbf{u} = \mu \Delta \mathbf{u}$.

With all this in mind, equation (8) yields equation (9).

$$\rho \frac{D\mathbf{u}}{Dt} = -\nabla (p - p_g) + \mu \Delta \mathbf{u} - \rho \beta (T - T_\infty) \mathbf{g} \tag{9}$$

Note that this equation is mainly used in order to reproduce problems with natural convection, as the effect of the body forces is relatively important. The Boussinesq term can be neglected for cases of forced convection, yielding equation (10).

$$\rho \frac{D\mathbf{u}}{Dt} = -\nabla p + \mu \Delta \mathbf{u} \tag{10}$$

2.3. Kinetic energy equation

The work-energy theorem states that the net work done by the forces on an object equals the change in its kinetic energy. Consequently, the kinetic energy equation is obtained by multiplying the linear momentum equation by the velocity of the fluid. Therefore, equation (8) is multiplied by $\mathbf{u}$ for an incompressible flow.

$$\rho \mathbf{u} \cdot \frac{D\mathbf{u}}{Dt} = -\mathbf{u} \cdot \nabla p + \mathbf{u} \cdot \nabla \cdot \boldsymbol{\tau} + \rho \mathbf{u} \cdot \mathbf{g} + \mathbf{u} \cdot \mathbf{f^e} \tag{11}$$

The different terms of equation (11) are analogous to the ones of the linear momentum equation, but now they represent the work done by the corresponding forces. At this moment, the following statements are taken into consideration:

- The kinetic energy is defined as $e_k = \mathbf{u} \cdot \mathbf{u}/2$. For this reason, the LHS of equation (11) turns into $\frac{De_k}{Dt} = \frac{1}{2}\frac{D(\mathbf{u}\cdot\mathbf{u})}{Dt} = \frac{1}{2}\left(\frac{D\mathbf{u}}{Dt} \cdot \mathbf{u} + \mathbf{u} \cdot \frac{D\mathbf{u}}{Dt}\right) = \mathbf{u} \cdot \frac{D\mathbf{u}}{Dt}$.
- The work of the pressure forces is re-expressed as $\mathbf{u} \cdot \nabla p = \nabla \cdot (p\mathbf{u}) - p \nabla \cdot \mathbf{u}$, and the work of the viscous forces[4] as $\mathbf{u} \cdot \nabla \cdot \boldsymbol{\tau} = \nabla \cdot (\mathbf{u} \cdot \boldsymbol{\tau}) - \boldsymbol{\tau} : \nabla \mathbf{u}$.

With this in mind, equation (11) turns into (12).

$$\rho \frac{De_k}{Dt} = -\nabla \cdot (p\mathbf{u}) + p \nabla \cdot \mathbf{u} + \nabla \cdot (\mathbf{u} \cdot \boldsymbol{\tau}) - \boldsymbol{\tau} : \nabla \mathbf{u} + \rho \mathbf{u} \cdot \mathbf{g} + \mathbf{u} \cdot \mathbf{f^e} \tag{12}$$

The electromagnetic forces are not of interest for the present dissertation. The term $\boldsymbol{\tau} : \nabla \mathbf{u}$ stands for the loss of kinetic energy due to the viscous dissipation. This is known as the Planck's postulate, defined in expression (13).

$$\boldsymbol{\tau} : \nabla \mathbf{u} \geq 0 \tag{13}$$

[4] Note that $\boldsymbol{\tau} : \nabla \mathbf{u} = \begin{bmatrix} \tau_{xx} & \tau_{xy} & \tau_{xz} \\ \tau_{yx} & \tau_{yy} & \tau_{yz} \\ \tau_{zx} & \tau_{zy} & \tau_{zz} \end{bmatrix} : \begin{bmatrix} \partial u/\partial x & \partial u/\partial y & \partial u/\partial z \\ \partial v/\partial x & \partial v/\partial y & \partial v/\partial z \\ \partial w/\partial x & \partial w/\partial y & \partial w/\partial z \end{bmatrix} = \tau_{xx}\frac{\partial u}{\partial x} + \tau_{xy}\frac{\partial u}{\partial y} + \tau_{xz}\frac{\partial u}{\partial z} + \tau_{yx}\frac{\partial v}{\partial x} + \tau_{yy}\frac{\partial v}{\partial y} + \tau_{yz}\frac{\partial v}{\partial z} + \tau_{zx}\frac{\partial w}{\partial x} + \tau_{zy}\frac{\partial w}{\partial y} + \tau_{zz}\frac{\partial w}{\partial z}$.

2.4. Total and thermal energy equations

The energy equation in its integral form, in an open system for a control volume in motion, is defined in equation (14). Note that the total energy is the sum of the internal energy and kinetic energy contributions, $e = u + e_k$ (per unit of density).

$$\frac{\partial}{\partial t}\int_{CV} e\rho \, dV + \oint_{CS} e\rho \left(\mathbf{u} - \mathbf{u_b}\right)\cdot \mathbf{n} \, dS = -\oint_{CS} \dot{\mathbf{q}}^{C+R}\cdot \mathbf{n} \, dS + \oint_{CS} \mathbf{u}\cdot \mathbf{f}_{(\mathbf{n})} \, dS + \int_{CV} \mathbf{u}\cdot \mathbf{g}\rho \, dV \quad (14)$$

The first term of the LHS stands for the change of the control volume's energy in time, and the second for the exchange of energy through its control surface. The first term of the RHS represents the energy change due to conduction and radiation heat, the second is the work of the surface forces, and the third the work of the gravity force.

Regarding the term corresponding to the work of the surface forces, it is expanded with the pressure and viscous forces for a static control volume. Performing an analysis over a differential control volume, equation (15) is obtained, which corresponds to the equation of transport of the total energy in differential notation.

$$\frac{\partial(\rho e)}{\partial t} + \nabla\cdot(\rho \mathbf{u} e) = -\nabla\cdot\dot{\mathbf{q}}^{C+R} - \nabla\cdot(p\mathbf{u}) + \nabla\cdot(\mathbf{u}\cdot\boldsymbol{\tau}) + \rho\mathbf{u}\cdot\mathbf{g} \quad (15)$$

The LHS can be expressed as $\frac{De}{Dt} = \frac{Du}{Dt} + \frac{De_k}{Dt}$, where the material derivative of the kinetic energy stands for equation (12). By doing so, equation (16) is obtained, corresponding to the equation of transport of the thermal energy for incompressible fluids.

$$\rho\frac{Du}{Dt} = -\nabla\cdot\dot{\mathbf{q}}^{C+R} - p\nabla\cdot\mathbf{u} + \boldsymbol{\tau} : \nabla\mathbf{u} \quad (16)$$

Assuming semi-perfect gas[5] and non-participating media (no heat flux due to radiation), equation (17) is obtained. The conduction heat transfer is considered with the Fourier's law $\dot{\mathbf{q}} = -\nabla\cdot(\lambda\nabla T)$.

$$\rho c_v\frac{DT}{Dt} = \nabla\cdot(\lambda\nabla T) - p\nabla\cdot\mathbf{u} + \boldsymbol{\tau} : \nabla\mathbf{u} \quad (17)$$

As the flow is incompressible, the divergence of the velocity is zero ($\nabla\cdot\mathbf{u} = 0$) and the heat capacity at constant volume is equal to the one at constant pressure ($c_v = c_p$). If the thermophysical properties are constant, the thermal conductivity gets out of the divergence operator of the Fourier's law. Furthermore, the assumption of negligible viscous dissipation is done, yielding equation (18). Note that $\frac{\lambda}{\rho c_v}$ corresponds to the thermal diffusivity α.

$$\frac{DT}{Dt} = \frac{\lambda}{\rho c_v}\nabla^2 T \quad (18)$$

The conduction heat flux always goes from high to low temperature zones. This is known as Fourier's postulate, stated in expression (19).

$$-\dot{\mathbf{q}}\cdot\nabla T \geq 0 \quad (19)$$

[5]For a semi-perfect gas, the internal energy is directly proportional to the temperature multiplied by the specific heat at constant volume $du = c_v dT$, with $c_v(T)$.

3. Generic Convection-Diffusion Equation

3.1. Theoretical aspects

All the Navier-Stokes equations can be summarized in a generic convection-diffusion equation, which is described in equation (20) for incompressible flow, two-dimensional Cartesian coordinates and constant physical properties.

$$\rho\frac{\partial\phi}{\partial t}+\rho u\frac{\partial\phi}{\partial x}+\rho v\frac{\partial\phi}{\partial y}=\Gamma\left(\frac{\partial^2\phi}{\partial x^2}+\frac{\partial^2\phi}{\partial y^2}\right)+S \tag{20}$$

The left-hand side stands for the transient and convective terms, while the right-hand side for the diffusive and source terms. In its convective compacted form, it takes the following form:

$$\rho\frac{\partial\phi}{\partial t}+\rho(\mathbf{u}\cdot\nabla)\phi=\Gamma\nabla^2\phi+S \tag{21}$$

For the sake of simplification, the divergence form will be employed henceforth. To obtain it, the condition of incompressibility ($\nabla\cdot\mathbf{u}=0$) must be taken into account[6].

$$\rho\frac{\partial\phi}{\partial t}+\rho\nabla\cdot(\mathbf{u}\phi)=\Gamma\nabla^2\phi+S \tag{22}$$

It is possible to create a table with the parameters to be replaced in the convection-diffusion equation to reproduce the Navier-Stokes equations in two-dimensions.

Equation	ϕ	Γ	S
Continuity	1	0	0
Momentum x-direction	u	μ	$-\partial p/\partial x$
Momentum y-direction	v	μ	$-\partial p/\partial y+\rho g\beta(T-T_\infty)$
Energy	T	λ/c_p	Φ/c_p

Table 1: Analogy between the parameters in the convection-diffusion equation with the Navier-Stokes equations.

3.2. Spatial discretization

To better understand the nomenclature, a schematic view of a generic control volume of the node-centered mesh is included in Figure 1.

[6]The following relation is used: $\nabla\cdot(\mathbf{u}\phi)=\phi(\nabla\cdot\mathbf{u})+(\mathbf{u}\cdot\nabla)\phi=(\mathbf{u}\cdot\nabla)\phi$, for incompressible flow.

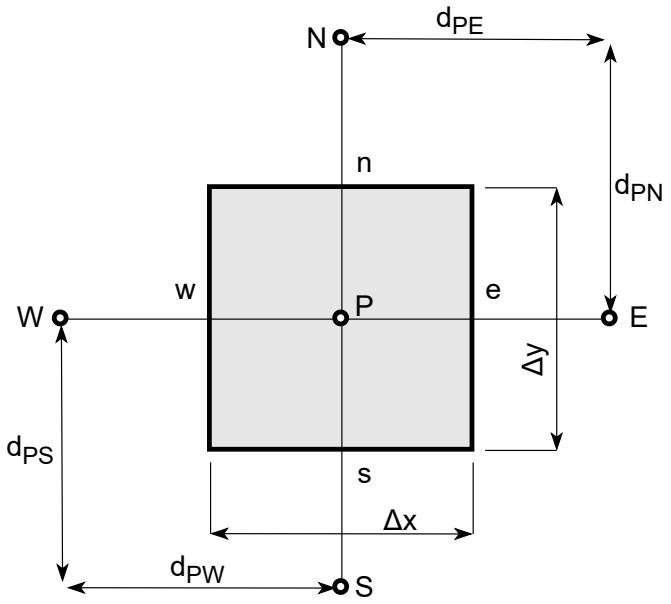


Figure 1: Illustration of a generic control volume of an arbitrary two-dimensional mesh.

The starting point for the spatial discretization of equation (22) is the *Gauss's divergence theorem*, which is stated as follows.

Theorem 1. *The integral of the divergence of a given vector* ($\mathbf{F}$) *over a closed volume in an arbitrary domain* (Ω)*, is equal to the flux of this vector across a closed surface enclosing the volume on its boundary* ($\partial\Omega$) *[4].*

$$\oint_{\Omega} (\nabla \cdot \mathbf{F})\, dV = \oint_{\partial\Omega} \mathbf{F} \cdot \mathbf{n}\, dS \tag{23}$$

Firstly, this theorem is applied to the diffusive term over the domain defined in Figure 1.

$$\oint_{\Omega} \nabla^2\phi\, dV = \oint_{\partial\Omega} \nabla\phi \cdot \mathbf{n}\, dS = \int_e \nabla\phi \cdot \mathbf{n}\, dS + \int_w \nabla\phi \cdot \mathbf{n}\, dS + \int_n \nabla\phi \cdot \mathbf{n}\, dS + \int_s \nabla\phi \cdot \mathbf{n}\, dS$$
$$\oint_{\Omega} \nabla^2\phi\, dV = \left.\frac{\partial\phi}{\partial x}\right|_e \Delta y - \left.\frac{\partial\phi}{\partial x}\right|_w \Delta y + \left.\frac{\partial\phi}{\partial y}\right|_n \Delta x - \left.\frac{\partial\phi}{\partial y}\right|_s \Delta x \tag{24}$$

The main concern for the resolution of equation (24) is the evaluation of the partial derivative of ϕ over the interfaces of the control volume. This is done by using the following second-order interpolation, based on a Taylor-series expansion around the corresponding face.

$$\left.\frac{\partial\phi}{\partial x}\right|_e = \frac{\phi_E - \phi_P}{d_{PE}} + O\left(\Delta x^2\right) \tag{25}$$

As it can be seen, dividing the length Δx by a factor of two decreases the error of the interpolation in a factor of four, due to the quadratic behaviour of this strategy.

Now, the theorem is applied to the convective term.

$$\oint_{\Omega} \nabla \cdot (\mathbf{u}\phi)\, dV = \oint_{\partial\Omega} (\mathbf{u}\phi) \cdot \mathbf{n}\, dS = \int_e u\phi\, dS - \int_w u\phi\, dS + \int_n v\phi\, dS - \int_s v\phi\, dS$$

$$\oint_{\Omega} \nabla \cdot (\mathbf{u}\phi)\, dV = (u\phi)_e \Delta y - (u\phi)_w \Delta y + (v\phi)_n \Delta x - (v\phi)_s \Delta x \tag{26}$$

Note that the velocity field in two dimensions is defined as $\mathbf{u} = [u;\, v]$. As it can be observed, the main concern in order to solve equation (26) is the evaluation of $(\mathbf{u}\phi)$ over the faces of the control volume.

Regarding the velocity, if the velocity field is known, the only thing to do is to evaluate it on the corresponding interface. For the evaluation of $(\phi)_e$, $(\phi)_w$, $(\phi)_n$ and $(\phi)_s$, there are different ways of approaching the problem by means of convective interpolation schemes. Some of them are the central difference, upwind, hybrid, power law and exponential schemes [5].

- *Central-difference scheme*: the convection term is computed as the mean value of the property evaluated on the two nearest nodes to the interface. For instance, at the east face:

$$\phi_e = \frac{1}{2}(\phi_E + \phi_P). \tag{27}$$

  The factor $\frac{1}{2}$ arises from the assumption that the interface is located midway between the nodes. Therefore, it acts as an interpolation factor that depends on the definition of the mesh. This form is the natural outcome of a second-order Taylor-series formulation. Depending on the value of the Péclet number at the interface of the elements, this scheme can turn the discretization coefficients negative, yielding to unrealistic results, instabilities and divergence of the solution. It can be proven that the Péclet number (P) shall be within the range of $-2 \leq P \leq +2$ to avoid the mentioned problems.

- *Upwind scheme*: the value of ϕ at an interface is equal to the value of ϕ at the upwind side of the face. Therefore, for the east face:

$$\begin{aligned} \phi_e &= \phi_P \quad \text{if} \quad F_e > 0, \\ \phi_e &= \phi_E \quad \text{if} \quad F_e < 0, \end{aligned} \tag{28}$$

  where the parameter $F_e = (\rho u)_e \Delta y$ corresponds to the mass flow rate on the east interface. The conditional statement can be rewritten as follows,

$$F_e\phi_e = \phi_P \cdot max[F_e,\, 0] - \phi_E \cdot max[-F_e,\, 0] \tag{29}$$

  As this strategy avoids negative coefficients, the solution will be always physically realistic. However, it is a first order scheme, so the solution will have greater error than on second order schemes.

- *Hybrid scheme*: this method is identical with the central-difference scheme when the Péclet number is within the range of $-2 \leq P \leq +2$. Outside this range, the scheme acts as an upwind scheme in which the diffusion is set to 0. Therefore, the solution will be always physically feasible, as the method uses the upwind scheme to avoid the problems of the central difference scheme. However, the error increases when $|P| \geq 2$.

- *Exponential scheme*: this method consists in employing the analytical exact solution of the convection-diffusion problem in the calculation of the coefficients. The mentioned exact solution is found in a steady one-dimensional case, with $0 \leq x \leq L$ as domain, yielding:

$$\frac{\phi - \phi_0}{\phi_L - \phi_0} = \frac{\exp\left(P \cdot x/L\right) - 1}{\exp\left(P\right) - 1}, \tag{30}$$

  where $\phi_0 = \phi(x = 0)$ and $\phi_L = \phi(x = L)$. Defining a total flux variable J in the x-direction as

$$J = \rho u \phi - \Gamma \frac{d\phi}{dx} \tag{31}$$

  and using the exact solution (30) for ϕ, employing ϕ_P, ϕ_E and $(\delta x)_e$ instead of ϕ_0, ϕ_L and L, it yields equation (32) for the east interface.

$$J_e = F_e \left(\phi_P + \frac{\phi_P - \phi_E}{\exp\left(P_e\right) - 1} \right) \tag{32}$$

  The procedure is analogous for the north, south and west faces. This solution can be introduced in equation (26) to obtain a second-order approach for the discretization coefficients.

- *Power law scheme*: the method consists on finding an approximation of the exponential behaviour described in the exact solution in equation (30). Although the expressions are relatively more complicated than for the hybrid scheme, they provide an extremely good representation of the exponential behaviour without a high computation cost.

3.3. Temporal discretization

There are two main ways of approaching the transient term of the convection-diffusion equation: the explicit and implicit temporal interpolation schemes. These are characterized by the following strategies.

$$\text{Explicit scheme:} \quad \frac{\partial \phi}{\partial t} \approx \frac{\phi^{n+1} - \phi^n}{\Delta t} = f\left(\phi^n,\ \phi^{n-1},\ \phi^{n-2},\ \phi^{n-3},\ ...\right)$$

$$\text{Implicit scheme:} \quad \frac{\partial \phi}{\partial t} \approx \frac{\phi^{n+1} - \phi^n}{\Delta t} = f\left(\phi^{n+1},\ \phi^n,\ \phi^{n-1},\ \phi^{n-2},\ ...\right)$$

On the one hand, the explicit scheme uses the already known values of ϕ, all the information comes from the past. It is a first-order scheme, not unconditionally stable with the value of Δt (which must be small enough to ensure the convergence of the solution). Despite this, an algorithm using this kind of scheme iterates faster than the ones using implicit schemes. It is widely used for highly convective problems, such as turbulent simulation.

On the other hand, the implicit scheme uses future values of ϕ. It is a first-order scheme unconditionally stable with the value of Δt. Despite iterating slower, the converged solution is usually obtained faster in this kind of scheme. It is widely used for highly diffusive problems, such as conduction heat transfer simulations.

There is a third strategy, which is a combination of the above. It is called Crank-Nicolson and corresponds to an unconditionally stable second-order scheme.

3.4. A 2-D Transient Conduction Problem

3.4.1. Problem definition

The section of a very long rod is composed of four different materials (M1, M2, M3 and M4), represented with different colours in Figure 2 below. This problem is a particular case of the generic convection-diffusion equation, as the conduction heat transfer is characterized by pure diffusion within a continuous medium.

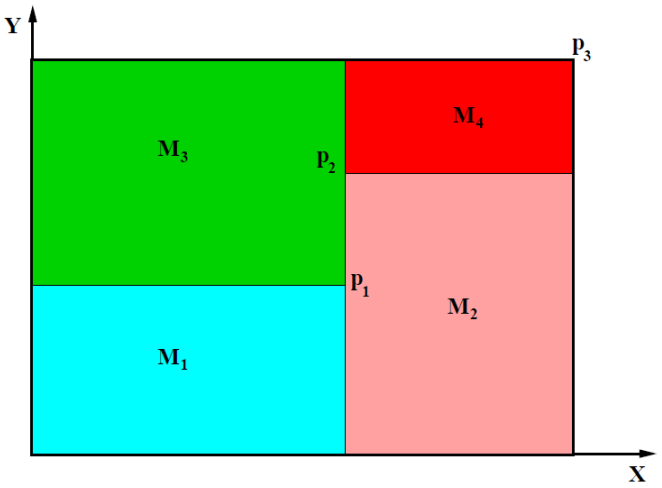


Figure 2: Four Materials. Illustration of the domain [6].

The coordinates of the points p_1, p_2 and p_3 are defined in Table 2.

	x [m]	y [m]
p_1	0.50	0.40
p_2	0.50	0.70
p_3	1.10	0.80

Table 2: Four Materials. Problem coordinates.

The properties of the different materials are given in Table 3. Note that the physical properties are constant in time, which will simplify the resolution algorithm.

	$\rho\ [kg/m^3]$	$c_p\ [J/kgK]$	$\lambda\ [W/Km]$
M_1	1,500	750	170
M_2	1,600	770	140
M_3	1,900	810	200
M_4	2,500	930	140

Table 3: Four Materials. Physical properties of the materials.

Each of the four sides of the rod interacts with the surrounding in a different manner, as described in Table 4. These are the boundary conditions of the problem.

Cavity wall	Boundary conditions
Bottom	Isotherm at $T = 23.00\ ^{\circ}C$
Top	Uniform $\dot{q}_{flow} = 60\ W/m$ length
Left	In contact with fluid at $T_g = 33.00\ ^{\circ}C$ and convective coefficient $h = 9.00\ W/m^2\ K$
Right	Uniform temperature $T = 8.00 + 0.005t\ ^{\circ}C$ (where t is the time in seconds)

Table 4: Four Materials. Boundary conditions.

The initial temperature field is $T = 8.00\ ^{\circ}C$. The main objective of this problem is to compute the temperature evolution of the rod, being the last instant of time equal to $10,000$ *sec*.

3.4.2. Discretized heat equation of pure diffusion

In this section, the conduction differential equations employed in solving the heat exchange through the rod are presented. Let's begin from the generic convection-diffusion equation (22), where the convective effects are non-existent as $\mathbf{u} = 0$ within this particular continuous medium.

$$\rho \frac{\partial \phi}{\partial t} = \Gamma \nabla^2 \phi + S \tag{33}$$

Particularizing equation (33) with the values of Table 1 applied to the energy equation:

$$\rho c_p \frac{\partial T}{\partial t} = \lambda \nabla^2 T + \Phi \tag{34}$$

Equation (34) is the heat equation of pure conduction, a direct consequence of the Fourier's law. The right hand side can be interpreted as the summation of heat flow entering or leaving a domain ($\sum \dot{Q}/V_P$), where the variable V_p stands for its volume. The left hand side is interpreted as the temporal variation in the internal energy of the same domain. This is also a particularization of the first law of thermodynamics. Therefore, discretizing this domain into an arbitrary number of control volumes of volume V_p each, equation (36) can be deduced.

$$\rho c_p \int_{\Omega} \frac{\partial T}{\partial t}\, dV = \int_{\Omega} \frac{\sum \dot{Q}}{V_P}\, dV \tag{35}$$

$$\rho V_P c_p \frac{\partial T}{\partial t} = \sum \dot{Q}_P \tag{36}$$

Note that, despite using the same nomenclature, ρ and c_p are now the thermophysical properties evaluated on the node P of the control volume. The right hand side stands for the summation of heat entering or leaving the arbitrary control volume. Discretizing in time, numerically integrating between the present and following instants, t^n and t^{n+1}:

$$\rho V_P c_p \int_{t^n}^{t^{n+1}} \frac{\partial T}{\partial t}\, dt = \int_{t^n}^{t^{n+1}} \sum \dot{Q}_P\, dt \tag{37}$$

$$\rho V_P c_p \left(T_P^{n+1} - T_P^n\right) = \left[\beta \sum \dot{Q}_P^{n+1} + (1-\beta) \sum \dot{Q}_P^n\right] \Delta t \tag{38}$$

Rearranging:

$$\frac{\rho V_P c_p}{\Delta t} T_P^{n+1} = \beta \sum \dot{Q}_P^{n+1} + (1-\beta) \sum \dot{Q}_P^{n} + \frac{\rho V_P c_p}{\Delta t} T_P^{n} \tag{39}$$

Where $\sum \dot{Q}_P^n$ and $\sum \dot{Q}_P^{n+1}$ represent the total heat summation at the present and following instants in the control volume, respectively, where the first is known at the moment of the calculation.

The variable β is a numerical parameter which defines the time numerical integration method as follows:

- $\beta = 0 \quad \longrightarrow \quad$ Explicit integration method: 1st order approach.
- $\beta = 0.5 \quad \longrightarrow \quad$ Crank-Nicolson integration method: 2nd order approach.
- $\beta = 1 \quad \longrightarrow \quad$ Implicit integration method: 1st order approach.

3.4.3. Discretization coefficients

First of all, a node-centered structured mesh over the domain is created by means of control volumes as defined in Figure 3. The parameters N and M define the number of elements in the x and y directions, respectively. Note that some boundary nodes are inserted in order to easily apply the boundary conditions.

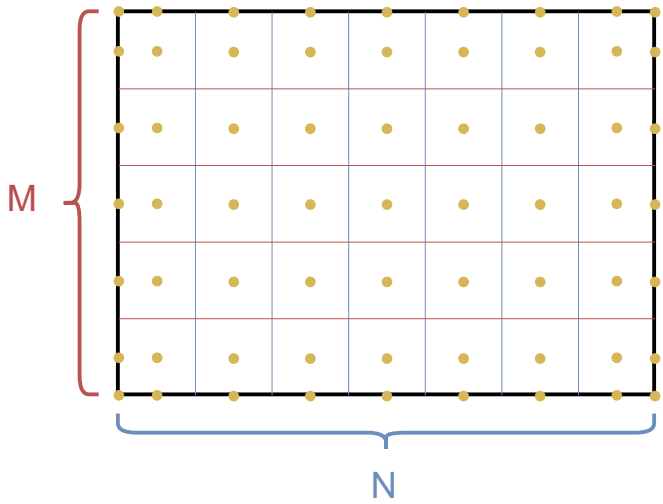


Figure 3: Four Materials. Illustration of the mesh discretization.

For each control volume, it is possible to obtain the summation of heat considering two-dimensional conduction heat flow between the nodes with the Fourier's law. Once the summations of heat at t^n and t^{n+1} are calculated, they are placed in equation (39), and then rearranged in order to obtain a set of linear equations with the following structure:

$$a_P T_P^{n+1} = a_E T_E^{n+1} + a_W T_W^{n+1} + a_S T_S^{n+1} + a_N T_N^{n+1} + b_P, \tag{40}$$

where the a_P, a_E, a_W, a_N, a_S and b_P are the discretization coefficients that multiply the nodal temperatures. For instance, the summation of heat at the instant t^{n+1} for an inner node of the mesh is stated in equation (41). Note that, as there is no additional source of heat in this problem, $\Phi = 0$.

$$\sum \dot{Q}_P^{n+1} = \lambda_w \frac{T_W^{n+1} - T_P^{n+1}}{d_{PW}} \Delta y - \lambda_e \frac{T_P^{n+1} - T_E^{n+1}}{d_{PE}} \Delta y + \lambda_s \frac{T_S^{n+1} - T_P^{n+1}}{d_{PS}} \Delta x - \lambda_n \frac{T_P^{n+1} - T_N^{n+1}}{d_{PN}} \Delta x \tag{41}$$

Replacing equation (41) into equation (39) and rearranging, it is possible to obtain the following discretization coefficients for the inner nodes.

$$\begin{cases} a_E = \beta \cdot \lambda_e \frac{\Delta y}{d_{PE}} \\ a_W = \beta \cdot \lambda_w \frac{\Delta y}{d_{PW}} \\ a_S = \beta \cdot \lambda_s \frac{\Delta x}{d_{PS}} \\ a_N = \beta \cdot \lambda_n \frac{\Delta x}{d_{PN}} \\ a_P = a_E + a_W + a_S + a_N + \frac{\rho V_P c_p}{\Delta t} \\ b_P = \frac{\rho V_P c_p}{\Delta t} T_P^n + (1-\beta) \sum \dot{Q}_P^n \end{cases} \tag{42}$$

The physical parameters are evaluated at the new instant t^{n+1}, which is not relevant in this problem since they are constant in time. The conductive heat transfer coefficient λ is evaluated at the corresponding face of the control volume by the use of an harmonic mean. For instance, λ_e is computed as:

$$\lambda_e = \frac{d_{PE}}{\frac{d_{Pe}}{\lambda_P} + \frac{d_{Ee}}{\lambda_E}}, \tag{43}$$

where d_{Pe} is the distance between the node P and the east face of the control volume, and analogous for d_{Ee}. At this point, the only thing left to do is to find the summation of heat at the boundary nodes, repeat the procedure and find the discretization coefficients of the nodes located on the walls.

Regarding the left and top walls, the summations of heat stand for equations (44) and (45), respectively. In regard to the nodes on the right and bottom walls, the temperature is already known for the imposed Dirichlet boundary conditions.

$$\sum \dot{Q}_P^{n+1} = h(T_g - T_P^{n+1}) \Delta y - \lambda_e \frac{T_P^{n+1} - T_E^{n+1}}{d_{PE}} \Delta y \tag{44}$$

$$\sum \dot{Q}_P^{n+1} = \dot{q}_{flow} \Delta x - \lambda_s \frac{T_P^{n+1} - T_S^{n+1}}{d_{PS}} \Delta x \tag{45}$$

The discretization coefficients can be now obtained by means of applying equation (39), yielding the results summarized in Table 5.

Boundary	Right wall	Left wall	Top wall	Bottom wall
a_E	0	$\beta\lambda_e\frac{\Delta y}{d_{PE}}$	0	0
a_W	0	0	0	0
a_N	0	0	0	0
a_S	0	0	$\beta\lambda_s\frac{\Delta x}{d_{PS}}$	0
a_P	1	$\beta\lambda_e\frac{\Delta y}{d_{PE}}+\beta h\Delta y$	$\beta\lambda_s\frac{\Delta x}{d_{PS}}$	1
b_P	T_{right}^{n+1}	$\beta hT_g\Delta y+(1-\beta)\sum\dot{Q}_P^n$	$\beta\cdot\dot{q}_{flow}\Delta x+(1-\beta)\sum\dot{Q}_P^n$	T_{bottom}

Table 5: Four Materials. Discretization coefficients of the boundary nodes.

Finally, note that the four nodes located at the edges of the domain are a particular case. Nonetheless, their temperatures can be perfectly avoided during the calculation, and assigned to the mean value of the temperature of their closest nodes at the end of the computation.

3.4.4. Algorithm

Firstly, the resolution of the linear system of equations defined by the set of equations (40) is carried out by a Gauss-Seidel algorithm. This consists on visiting each node iteratively, computing the variable T_P^{n+1} from equation (40) using the latest updated value for the temperature of the rest of the nodes. This method is also known as the *successive displacement method*, and it is an improved form of the Jacobi method.

The difference between both algorithms is that the Jacobi method uses always the values obtained in the previous iteration, while the Gauss-Seidel method always applies the latest updated values during the iterative process. More details are given in reference [7]. Regarding the global algorithm employed in solving the problem, it is defined below.

1. **Input data.**
   a) Coordinates of the points p_1, p_2 and p_3.
   b) Thermophysical variables of the four materials.
   c) Boundary conditions and initial map of temperatures (T_{ini}).
   d) Numerical parameters: mesh, time increment (Δt), maximum time (t_{max}), integration method (β), relaxation factor and convergence tolerance (δ).
   e) Coordinates of two points, if desired, to calculate the temperature evolution on them.
   f) Time on which saving the temperature distribution for post-processing isotherm plot.
2. **Previous calculus.**
   a) Structured mesh discretization and sizing of each volume of control.
   b) Generate a matrix with the number of the material associated to each element.
   c) Look for the closer elements to the two points on which to calculate the evolution of their temperatures.
   d) Initialization of the temperature matrices.
      i. $T^n = T_{ini}$ for the initial calculation temperature.
      ii. $T^* = T_{ini}$ as an auxiliary matrix for the Gauss-Seidel algorithm.
   e) Initialization of the time variable $t^n = 0\ sec$ and $t^{n+1} = \Delta t\ sec$.
3. **Loop.**
   a) Coefficient calculation: just one time per iteration, as the thermophysical variables are constant.

b) Gauss-Seidel algorithm.
   i. Apply equation (40) from the left to right, from the bottom to the top side nodes to calculate T_P^{n+1}, using T^* when the temperature is unknown and the latest calculated temperatures when possible.
   ii. Is $max|T^{n+1} - T^*| < \delta$?
      A. If yes, go to (3c).
      B. If not, $T^* = T^{n+1}$ and go to (3bi).

c) Maximum time condition. Is $t < t_{max}$?
   i. If yes, $t^n = t^{n+1}$, $t^{n+1} = t^{n+1} + \Delta t$, $T^n = T^{n+1}$. Save variables of interest. Go to (3a).
   ii. If not, go to (4).

4. **Post-process.**
   a) Plotting of the evolution of temperature on the two specified points.
   b) Representation of the isotherms map at the specified time with MATLAB.

In order to summarize the algorithm above, a flux diagram is included in Figure 4.

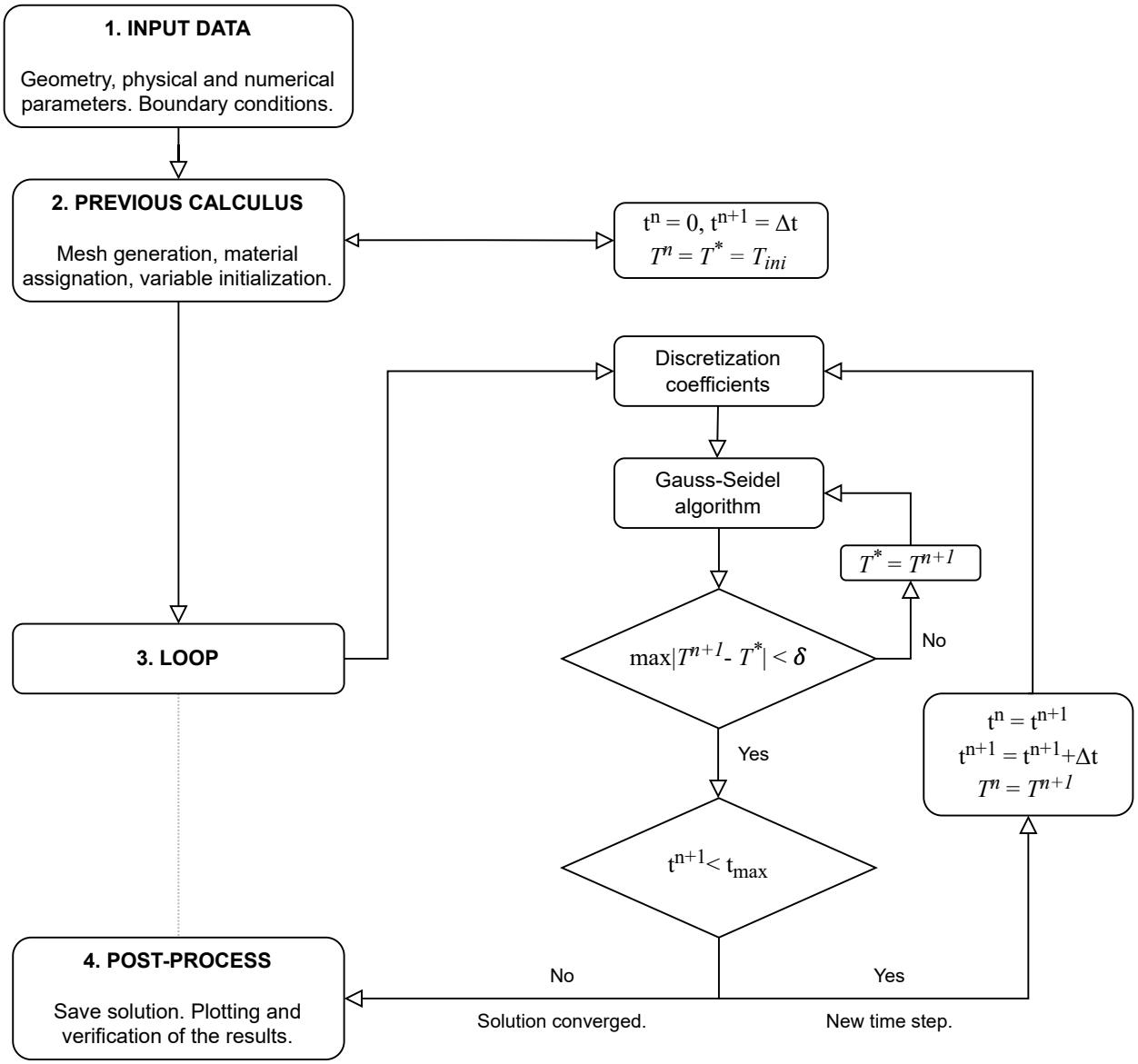
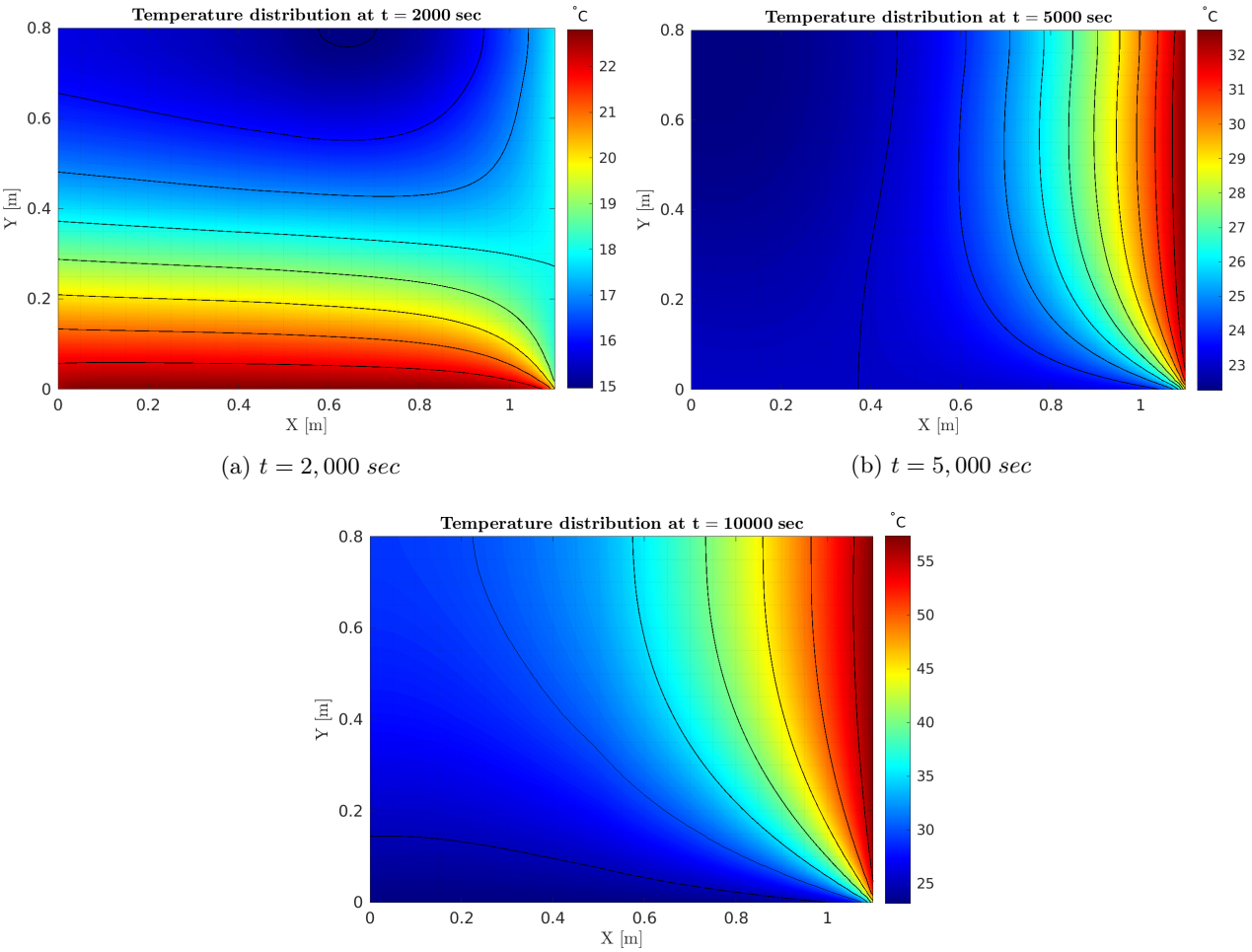


Figure 4: Four Materials. Resolution algorithm.

3.4.5. Results

First of all, the temperature distribution along the section of the rod is represented at different time instants from $0 \leq t \leq 10,000$ *sec*. Figure 5 are instantaneous plots at $t = 2,000$ *sec*, $t = 5,000$ *sec* and $t = 10,000$ *sec* obtained by means of a 110×80 element mesh with a Crank-Nicolson scheme.



(a) $t = 2,000$ *sec*

(b) $t = 5,000$ *sec*

(c) $t = 10,000$ *sec*

Figure 5: Four Materials. Temperature distribution at different instants.

The simulation starts from an initial temperature field of 8.00 $°C$ and the boundary conditions defined in Table 4. The bottom wall has a constant temperature of 23.0 $°C$, which is the higher temperature until the time instant $t = 3,000$ *sec*, when the temperature of the bottom and right walls gets to be equal. Figure 5a shows the isotherms at $t = 2,000$ *sec*, before the mentioned instant, when the bottom wall is the main source of heat.

As the time keeps increasing, the temperature of the right wall gets higher. This can be seen in Figures 5b and 5c, standing for $t = 5,000$ *sec* and $t = 10,000$ *sec*, respectively. The right wall is the main source of heat when the time increases, producing high temperature gradients along the domain as the bottom wall keeps its temperature constant.

These results can be compared to the ones given by the CTTC in order to verify the code, which are included in Figure 6. This must be compared to Figure 5b since both figures illustrate the same instant of time ($t = 5,000\ sec$). As it can be seen, the results of the reference coincide to the ones of the simulation. Therefore, the code is considered to work properly.

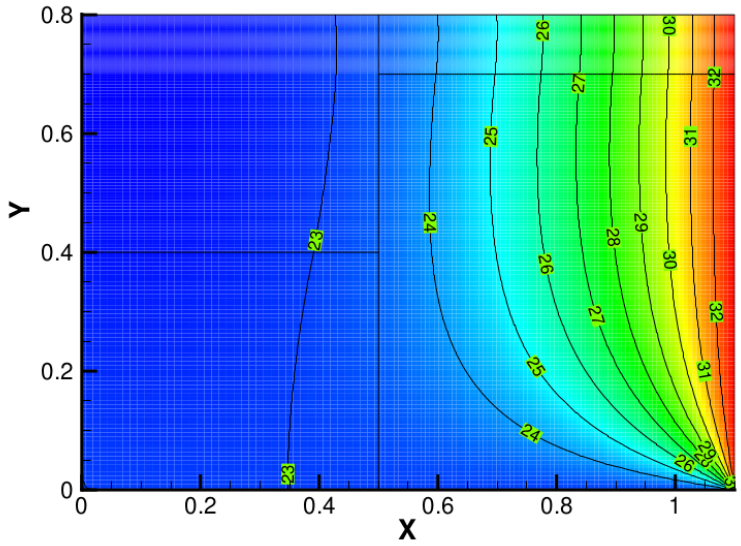


Figure 6: Four Materials. Reference temperature distribution at $t = 5,000\ sec$ [6].

In regard to the temperature evolution in time of the coordinates $P_1 = [0.65,\ 0.56]$ and $P_2 = [0.74,\ 0.72]$, Figure 7 is included.

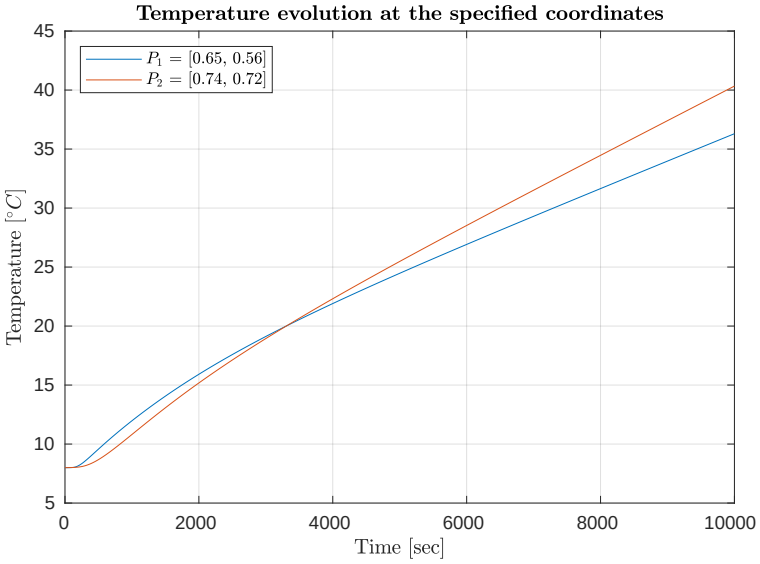


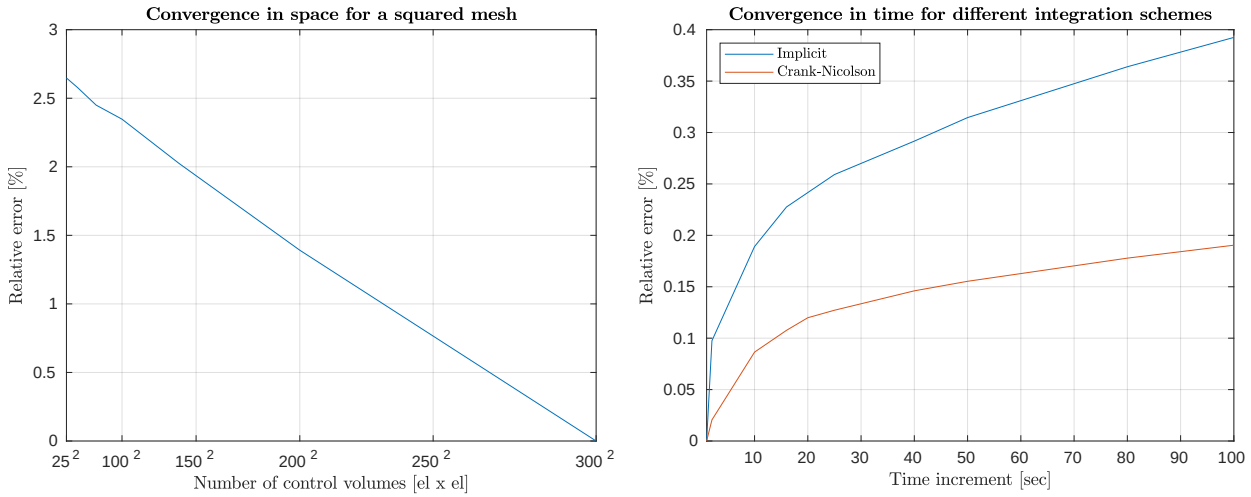
Figure 7: Four Materials. Temperature evolution in time at P_1 and P_2.

It can be observed that the temperature of both points tends to increase in time. This is caused by the fact that the temperature of the right wall gets higher as the simulation advances, introducing more heat to the domain.

Finally, the convergence of the code in terms of the mesh size and time step is analyzed. On the one hand, to understand the convergence of the results regarding the size of the mesh, the relative variation of the temperature on the northwest corner at the instant $t = 10,000$ *sec* is represented against different mesh sizes for a squared mesh. The top-left corned is used for this computation, as there is always a fixed node on this position regardless the size of the mesh. This can be seen in Figure 8a.

On the other hand, the convergence of the results regarding the time increment is studied for a 110×80 element mesh. This is done representing the relative variation of the temperature at the northwest corner at the instant $t = 10,000$ *sec* against the time increment employed in the simulation. The following Figure 8b is obtained.

Note that, since there is no analytical solution for this problem, the relative error is referenced to the most refined case (a mesh of 300×300 elements and a time step of 1 second).



(a) Mesh size analysis.

(b) Time increment analysis.

Figure 8: Four Materials. Analysis of the convergence of the results in space and time.

As it is possible to observe, using a squared mesh with 150 elements on each direction gives a relative error of less than 2.0%. Regarding the time step, using $\Delta t = 100$ *sec* gives relative errors of less than 0.5%. Despite this, as the computing time is relatively low, it is possible to employ a much bigger mesh and a lower time step.

Finally, note that the relative error on the spatial discretization decreases quadratically with the number of elements on each direction, and the Crank-Nicolson gives less error than the implicit scheme.

3.5. The Smith-Hutton Problem

3.5.1. Problem definition

The objective is to find the steady state solution of the Smith-Hutton problem by means of solving numerically a generic convection-diffusion equation, presented in equation (22). In order to do so, the Finite Volume Method is applied to the rectangular domain described in Figure 9.

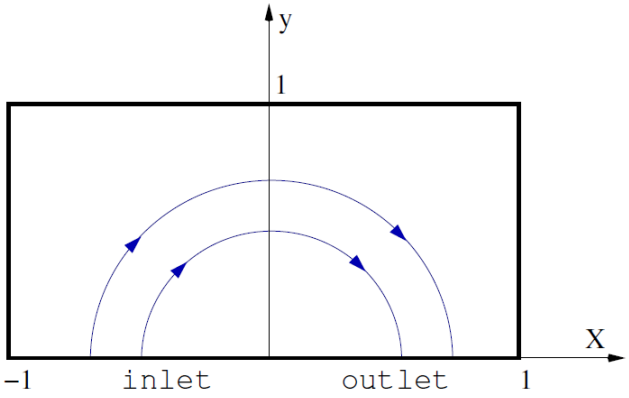


Figure 9: Smith-Hutton. General schema of the proposed problem [8].

A prescribed velocity field is given as follows:

$$\begin{aligned} u(x,y) &= \quad 2y(1-x^2), \\ v(x,y) &= -2x(1-y^2). \end{aligned} \tag{46}$$

The boundary conditions for the variable ϕ are described below, where $\alpha = 10$.

$$\begin{aligned} \phi &= 1 + \tanh\left(\alpha(2x+1)\right) && y = 0 \;\; ; \;\; x \in (-1,0) && \text{(inlet)} \\ \frac{\partial \phi}{\partial y} &= 0 && y = 0 \;\; ; \;\; x \in (0,+1) && \text{(outlet)} \\ \phi &= 1 - \tanh(\alpha) && \text{(elsewhere)} \end{aligned} \tag{47}$$

3.5.2. Practical aspects

In order to solve equation (22), the integral in time is computed numerically obtaining equation (49).

$$\int_{t^n}^{t^{n+1}} \frac{\partial(\rho\phi)}{\partial t} dt = \int_{t^n}^{t^{n+1}} \left(\nabla \cdot (\Gamma \nabla \phi) - \nabla \cdot (\rho \mathbf{u} \phi) + S\right) dt = \int_{t^n}^{t^{n+1}} I dt \tag{48}$$

$$(\rho\phi)_P^{n+1} - (\rho\phi)_P^n = \left[\beta I^{n+1} + (1-\beta) I^n\right] \Delta t \tag{49}$$

The value of the parameter β defines if the numerical integration follows an explicit ($\beta = 0$), Crank-Nicolson ($\beta = 0.5$) or implicit ($\beta = 1$) scheme. Note that the integrand has been defined as $I = \nabla \cdot (\Gamma \nabla \phi) - \nabla \cdot (\rho \mathbf{u} \phi) + S$.

Discretizing this integrand taking into account the Finite Volume Method, equation (50) is obtained, which corresponds to the fully discretized form of the convection-diffusion equation that is solved in the present dissertation.

$$\begin{aligned}\frac{(\rho\phi)_P^{n+1}-(\rho\phi)_P^n}{\Delta t}\Delta x\Delta y+\beta\left[(\rho u\phi)_e^{n+1}-(\rho u\phi)_w^{n+1}\right]\Delta y+\beta\left[(\rho v\phi)_n^{n+1}-(\rho v\phi)_s^{n+1}\right]\Delta x\\ =\beta\left[\left(\Gamma\frac{\phi_E-\phi_P}{d_{PE}}\right)^{n+1}-\left(\Gamma\frac{\phi_P-\phi_W}{d_{PW}}\right)^{n+1}\right]\Delta y+\beta\left[\left(\Gamma\frac{\phi_N-\phi_P}{d_{PN}}\right)^{n+1}-\left(\Gamma\frac{\phi_P-\phi_S}{d_{PS}}\right)^{n+1}\right]\Delta x\\ +\beta S_P^{n+1}\Delta x\Delta y+(1-\beta)I^n\Delta x\Delta y\end{aligned} \tag{50}$$

The aforementioned convective schemes yield to the following discretization equations:

$$a_P\phi_P^{n+1}=a_E\phi_E^{n+1}+a_W\phi_W^{n+1}+a_S\phi_S^{n+1}+a_N\phi_N^{n+1}+b_P \tag{51}$$

Where the coefficients can be calculated as follows.

$$\begin{cases}a_E=\beta D_eA(|P_e|)+max[-\beta F_e,\,0],\\ a_W=\beta D_wA(|P_w|)+max[+\beta F_e,\,0],\\ a_N=\beta D_nA(|P_n|)+max[-\beta F_n,\,0],\\ a_S=\beta D_sA(|P_s|)+max[+\beta F_s,\,0],\\ a_P=a_E+a_W+a_N+a_S+\rho^{n+1}\frac{\Delta x\Delta y}{\Delta t},\\ b_P=\beta S_P^{n+1}\Delta x\Delta y+(1-\beta)I^n\Delta x\Delta y+\rho^n\frac{\Delta x\Delta y}{\Delta t}\phi_P^n.\end{cases} \tag{52}$$

The function $A(|P|)$ depends on the employed scheme and is obtained by the use of Table 6.

Scheme	Formula for $A(\|P\|)$
Central-difference (CDS)	$1-0.5\|P\|$
Upwind (UDS)	1
Hybrid (HDS)	$\max[0,\ 1-0.5\|P\|]$
Power law (PLDS)	$\max[0,\ (1-0.1\|P\|)^5]$
Exponential (EDS)	$\|P\|/[\exp(\|P\|)-1]$

Table 6: Smith-Hutton. Function $A(|P|)$ for different convective schemes.

Note that for the CDS, $A(|P|)$ becomes negative within the $-2\leq P\leq+2$ range, yielding the aforementioned solver instabilities. Therefore, this situation must be avoided.

The corresponding conductances (D) and mass flow rates (F) through the faces of the control volume are defined by the following expressions.

$$\begin{aligned} D_e &= \frac{\Gamma_e \Delta y}{d_{PE}} & F_e &= (\rho u)_e \Delta y \\ D_w &= \frac{\Gamma_w \Delta y}{d_{PW}} & F_w &= (\rho u)_w \Delta y \\ D_n &= \frac{\Gamma_n \Delta x}{d_{PN}} & F_n &= (\rho v)_n \Delta x \\ D_s &= \frac{\Gamma_s \Delta x}{d_{PS}} & F_s &= (\rho v)_s \Delta x \end{aligned} \tag{53}$$

Finally, the Péclet numbers are described by:

$$P_e = \frac{F_e}{D_e}, \qquad P_w = \frac{F_w}{D_w}, \qquad P_n = \frac{F_n}{D_n}, \qquad P_s = \frac{F_s}{D_s}. \tag{54}$$

3.5.3. Algorithm

An algorithm is programmed in C++ language in order to find the steady solution by means of the evolution of a transient system. Note that with the aim of simplifying the problem, only the implicit numerical integration has been programmed.

1. **Input data.**
   a) Geometry parameters such as the definition of the limits of the domain.
   b) Physical parameters. Relation between ρ and Γ. Definition of the velocity field equations for u and v.
   c) Numerical parameters: mesh, time increment (Δt), maximum time (t_{max}), integration method ($\beta = 1$), relaxation factor and convergence tolerance (δ), initial value for the scalar field (ϕ_{ini}).
   d) Arbitrary variable for setting the convective interpolation scheme.
2. **Previous calculus.**
   a) Uniform geometry discretization and sizing of each volume of control.
   b) Initialization of the scalar field matrices.
      i. $\phi^n = \phi_{ini}$ for the initial calculation.
      ii. $\phi^* = \phi^n$ as an auxiliary matrix.
   c) Initialization of time variable $t^n = 0\ sec$ and $t^{n+1} = \Delta t\ sec$.
3. **Loop.**
   a) Coefficient calculation: just one time per iteration, as the thermophysical variables are constant. The boundary conditions for ϕ are imposed.
   b) Gauss-Seidel algorithm.
      i. Apply equation (51) from the left to right, from the bottom to the top side nodes to calculate ϕ_P^{n+1}, using ϕ^* when the value of the scalar field is unknown and the last calculated value of ϕ when possible.
      ii. Is $max|\phi^{n+1} - \phi^*| < \delta$?
         A. If yes, go to (3c).
         B. If not, $\phi^* = \phi^{n+1}$ and go to (3bi).
   c) Maximum time condition. Is $t^{n+1} < t_{max}$?

i. If not, the solution is not converged within the specified maximum time. The loop is stopped and an error message is displayed on the screen.
ii. If yes, is $max|\phi^{n+1} - \phi^n| < \delta$?
   A. If not, $t^n = t^{n+1}$, $t^{n+1} = t^{n+1} + \Delta t$, $\phi^n = \phi^{n+1}$. Go to (3a).
   B. If yes, the solution is converged and the loop stopped. A message is displayed on the screed to notify it. Go to (4).

4. **Post-process.**
   a) Plotting of the evolution of the variable ϕ in the outlet-inlet bottom side, and comparison of the results with the reference values.
   b) Plotting of the contour map of the scalar field at the specified time t_{max} with MATLAB.

In order to summarize the aforementioned algorithm, a flux diagram is included in Figure 10.

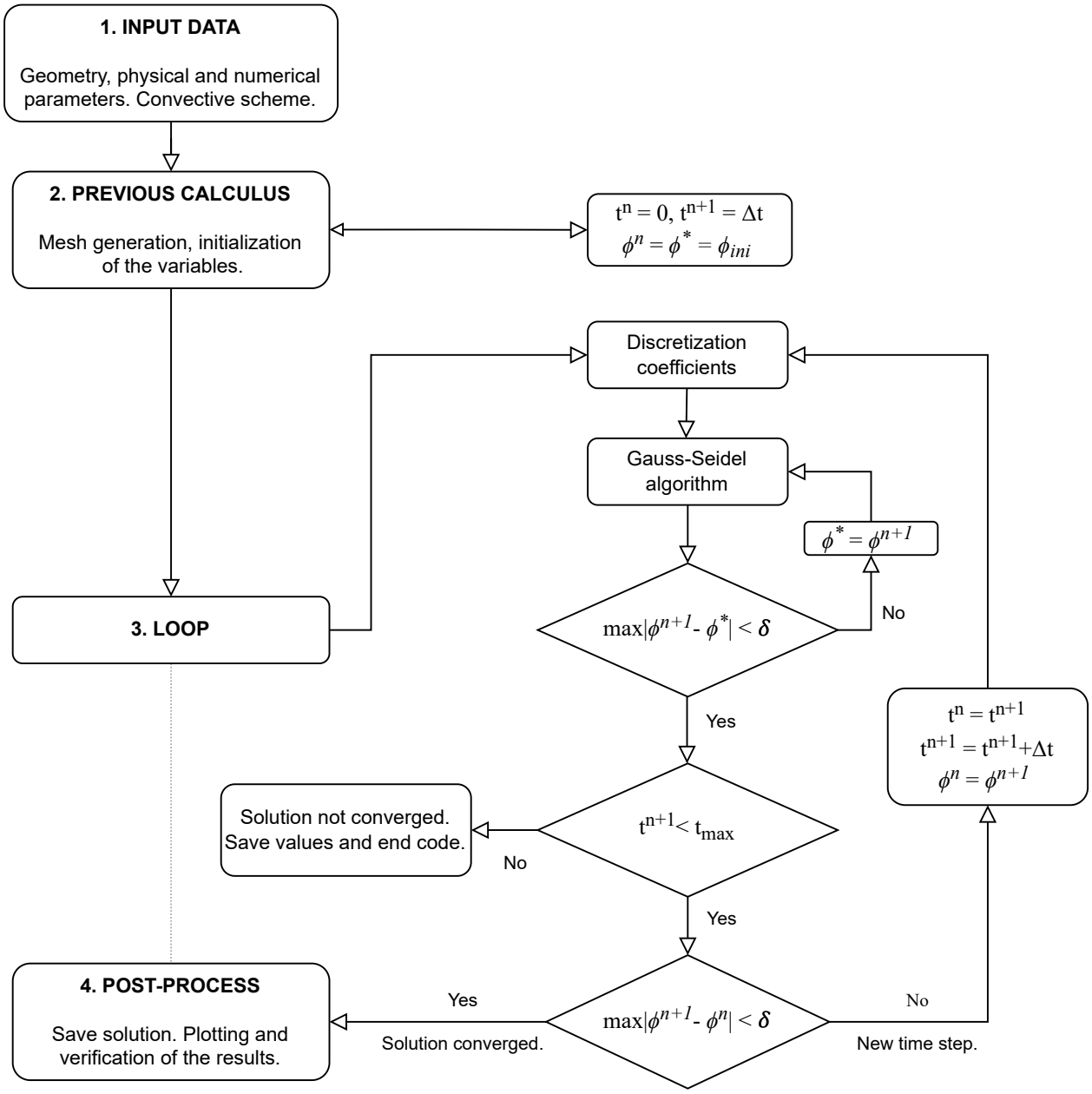


Figure 10: Smith-Hutton. Resolution algorithm.

3.5.4. Results

The simulation is carried out by means of a first order Upwind Scheme (UDS), with a mesh of 600×300 elements. The reason for this choice is to avoid instabilities in the results for high ratios of ρ/Γ, as will be explained later in the present section. Table 7 is a summary of the obtained results for different values of ρ/Γ, which can be compared with reference [8] in Table A.1 of the Appendix.

x-position	ϕ_{outlet} $\rho/\Gamma = 10$	ϕ_{outlet} $\rho/\Gamma = 10^3$	ϕ_{outlet} $\rho/\Gamma = 10^6$
0.0	2.000	2.000	2.000
0.1	1.402	2.000	2.000
0.2	1.143	1.999	2.000
0.3	0.942	1.991	1.999
0.4	0.769	1.807	1.956
0.5	0.614	0.910	0.886
0.6	0.472	0.126	0.025
0.7	0.341	0.004	0.001
0.8	0.218	0.000	0.000
0.9	0.102	0.000	0.000
1.0	0.000	0.000	0.000

Table 7: Smith-Hutton. Summary of the simulation results.

First of all, in order to understand the instabilities that appear when using a second order CDS scheme, Figures 11 and 12 are included for $\rho/\Gamma = 10$ and $\rho/\Gamma = 10^3$, respectively. They represent the value of ϕ at $y = 0\ m$, using a CDS scheme with 100×50 elements. The reference values are superposed over the simulation values in order to compare the plots with the reference.

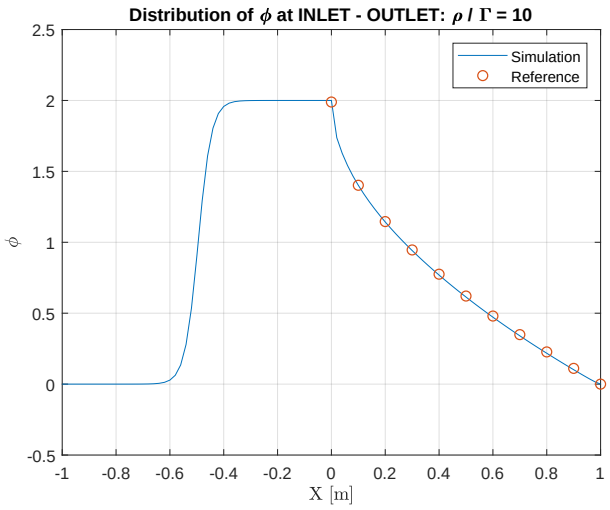


Figure 11: Smith-Hutton. Comparison between the results and the reference values at the INLET-OUTLET for $\rho/\Gamma = 10$.

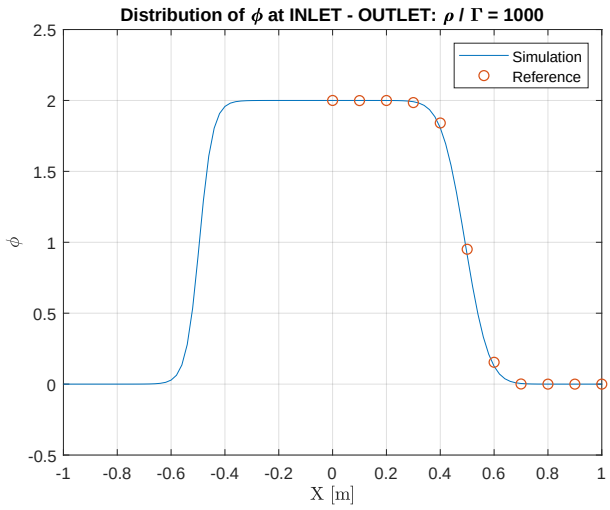
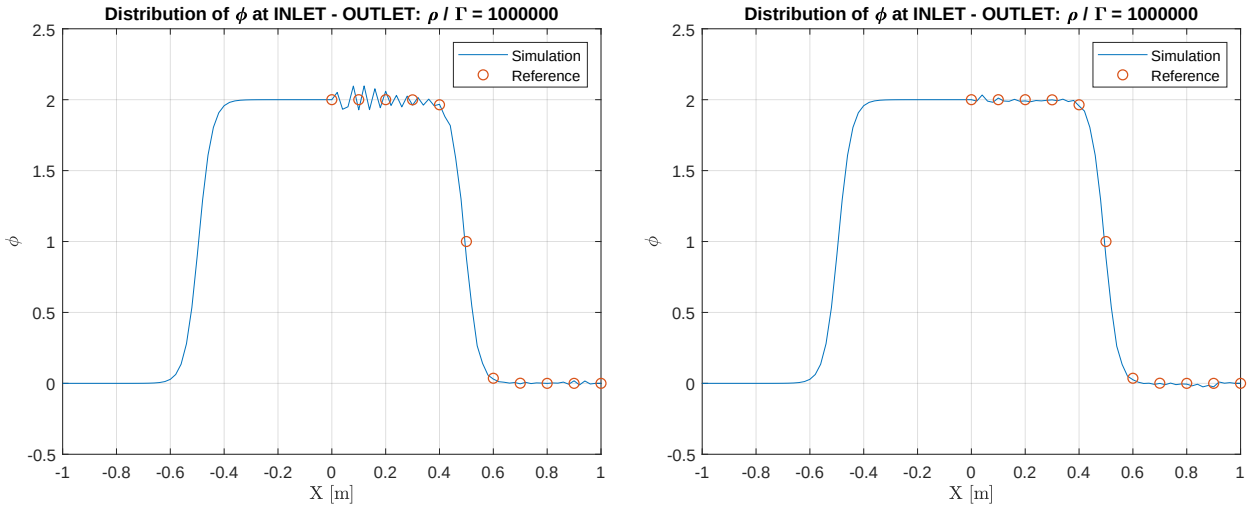


Figure 12: Smith-Hutton. Comparison between the results and the reference values at the INLET-OUTLET for $\rho/\Gamma = 10^3$.

It is possible to observe that, for the aforementioned cases, the behaviour between the simulation and reference values is very similar and the results coincide without great error. Therefore, the instabilities caused by the fact of using the CDS scheme do not appear at low values of ρ/Γ.

Otherwise, for $\rho/\Gamma = 10^6$ the Péclet numbers can turn greater than $|P| \geq 2$, and several instabilities appear when the time increases, yielding to a fatal error before finding any steady state solution. This is caused by the fact that, when the relative weight of the convective effects is much greater than the diffusion effects, the Central Difference Scheme can not represent well sharpen gaps on the value of ϕ.

This effect is known as numerical diffusion, which can cause oscillations on the solution as can be seen in Figure 13a. Despite the solution is unstable, it is possible to obtain a partially converged solution before the appearance of the mentioned instabilities, as shown in Figure 13b.



(a) Oscillations caused by the CDS.

(b) Results before the instabilities.

Figure 13: Smith-Hutton. Instabilities caused by the CDS at the INLET-OUTLET and $\rho/\Gamma = 10^6$.

Although being a first-order scheme, the UDS avoids these instabilities by definition. For this reason, it is convenient to use an Upwind Scheme with greater mesh size when the diffusive effects are low and the variable ϕ changes a lot in short distances. In order to study the necessary mesh size when using a first-order scheme such as UDS, Figure 14 is included. As Table 7 uses the same scheme and mesh size for all the values of ρ/Γ, it is convenient to evaluate Figure 14 on $\rho/\Gamma = 10^3$.

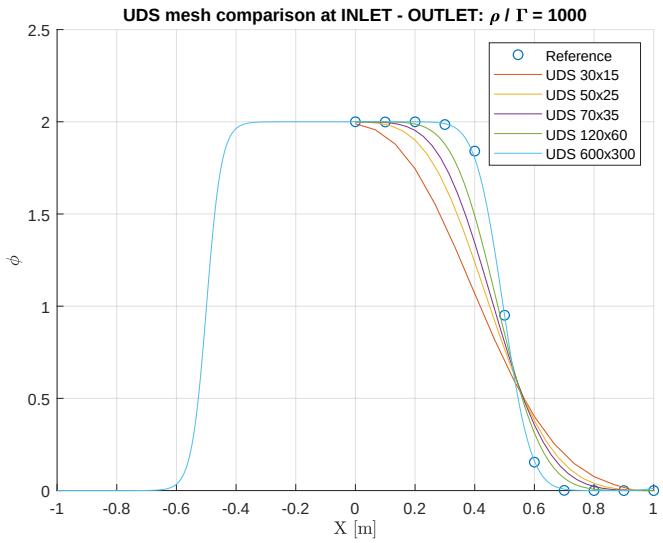


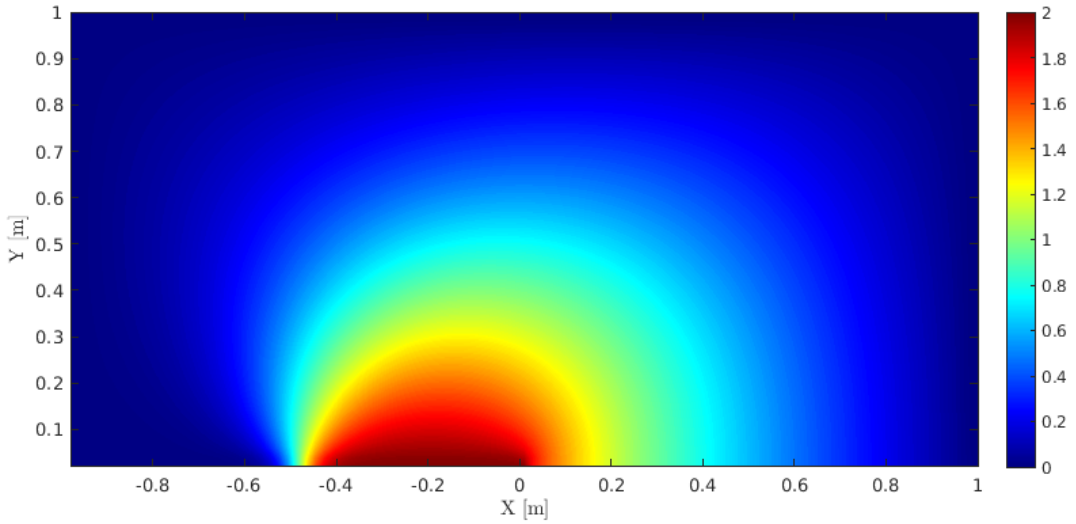
Figure 14: Smith-Hutton. UDS results comparison for different mesh sizes at $\rho/\Gamma = 10^3$.

At it can be seen, while a second-order scheme (such as CDS) converges well on the solution with a mesh of 100×50 elements, a first-order scheme (such as UDS) needs a denser mesh of 600×300 elements, for instance.

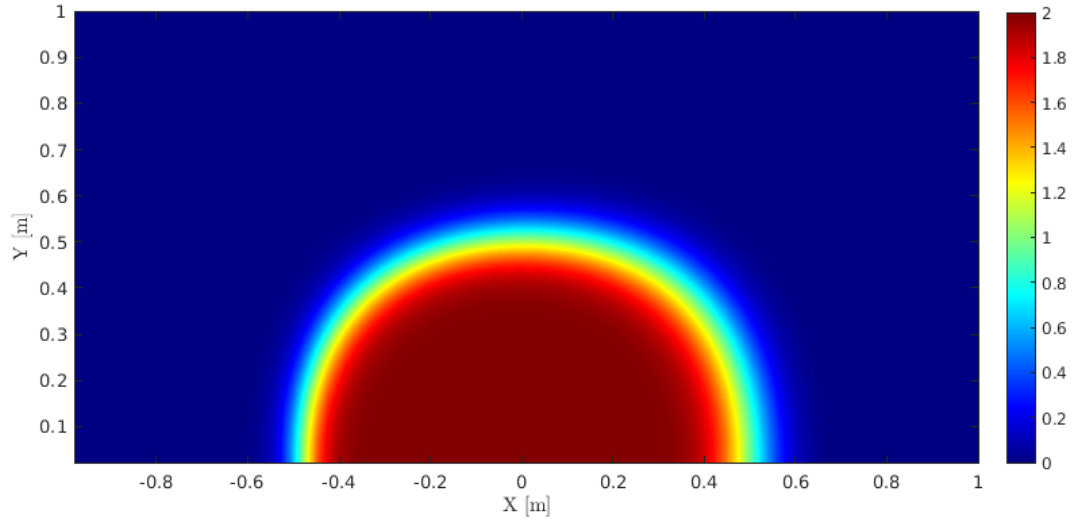
Finally, a map of isotherms is included for each studied case in Figure 15. Note that the parameter ρ/Γ can be understood as the ratio between convective and diffusive effects. When this ratio is low, the diffusive effects are very strong. Otherwise, for high values of ρ/Γ, the transport of the variable is ϕ is mainly caused by convective effects.

On the one hand, Figure 15a is an example of high diffusive transport of ϕ. The solution is smoothed along the domain, as the sharpened gaps of the variable ϕ in the inlet do not appear in the outlet. This variable tends to follow the velocity field, transporting its value from high to low energized zones (from high to low values of ϕ) by means of a physical diffusion process.

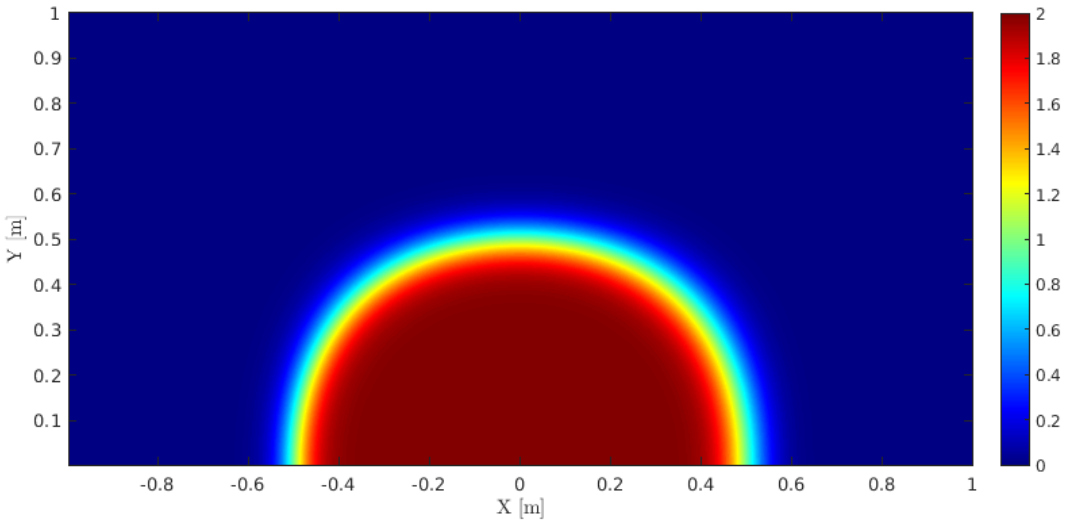
On the other hand, Figure 15c is an example of high convective transport of ϕ. The diffusive effects are low, and the transport mechanism of the variable ϕ is mainly the velocity field. The solution tends to be symmetric over the vertical line $x = 0\ m$. Finally, despite the convective effects are more notable than the diffusive ones, Figure 15b is a combination of both.



(a) $\rho/\Gamma = 10$



(b) $\rho/\Gamma = 10^3$



(c) $\rho/\Gamma = 10^6$

Figure 15: Smith-Hutton. Map of isotherms at the steady solution, using a UDS 600x300.

4. The Fractional Step Method

The theoretical aspects of the Fractional Step Method (FSM) are explained in this section. This method allows to solve simultaneously the incompressible momentum and mass conservation equations, bonded by the gradient of the pressure field.

4.1. Incompressible Navier-Stokes equations

The starting point are the governing equations of the fluid. For the present dissertation, these are the dimensionless Navier-Stokes equations in their compacted vectorial form, defined in equations (55) and (56). The first stands for the momentum equation, and the second for the mass conservation of an incompressible flow. Note that the momentum equations are a set of convection-diffusion equations bonded to the mass conservation by means of their source term (the gradient of the pressure field).

$$\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla)\mathbf{u} = \frac{1}{Re}\Delta \mathbf{u} - \nabla p \tag{55}$$

$$\nabla \cdot \mathbf{u} = 0 \tag{56}$$

The Reynolds number is defined as $Re = \frac{\rho V_0 L}{\mu}$, where ρ and μ are the density and viscosity of the fluid. V_0 and L are the characteristic velocity and length, respectively.

4.2. Helmholtz-Hodge Decomposition theorem

The Helmholtz-Hodge Decomposition theorem (HHD) is essential for the numerical approximation of the incompressible Navier-Stokes equations [9]. It is stated as follows:

Theorem 2. *A vector field $\vec{\omega}$, defined in a bounded domain Ω with smooth boundary $\partial\Omega$, is uniquely determined when its divergence $(\nabla \cdot \vec{\omega})$ and rotational $(\nabla \times \vec{\omega})$ are assigned along with the normal $(\omega_n = \vec{n} \cdot \vec{\omega})$ or tangential $(\vec{\omega}_t = \vec{n} \times \vec{\omega})$ component on the boundary*

$$\vec{\omega} = \nabla D + \nabla \times \vec{R} = \vec{d} + \vec{r} \tag{57}$$

where

$$\nabla \cdot \vec{d} = \nabla \cdot \vec{\omega} = \Delta D \tag{58}$$

$$\nabla \times \vec{r} = \nabla \times \vec{\omega} = \nabla \times (\nabla \times \vec{R}) \tag{59}$$

satisfying the following boundary conditions:

$$\vec{n} \cdot \vec{r} = 0 \longrightarrow \vec{n} \cdot \nabla D = \omega_n \tag{60}$$

$$\vec{n} \times \vec{d} = 0 \longrightarrow \vec{n} \times (\nabla \times \vec{R}) = \vec{\omega}_t \tag{61}$$

with $\vec{n}$ being normal to the boundary. Therefore, the vector field $\vec{w}$ can be uniquely expressed as the sum of an irrotational $(\vec{d})$ and an incompressible $(\vec{r})$ contribution.

An illustration of the aforementioned contributions is shown in Figure 4.2. In the case of the Navier-Stokes equations, the gradient field corresponds to the pressure gradient, and the divergence-free field stands for the velocity.

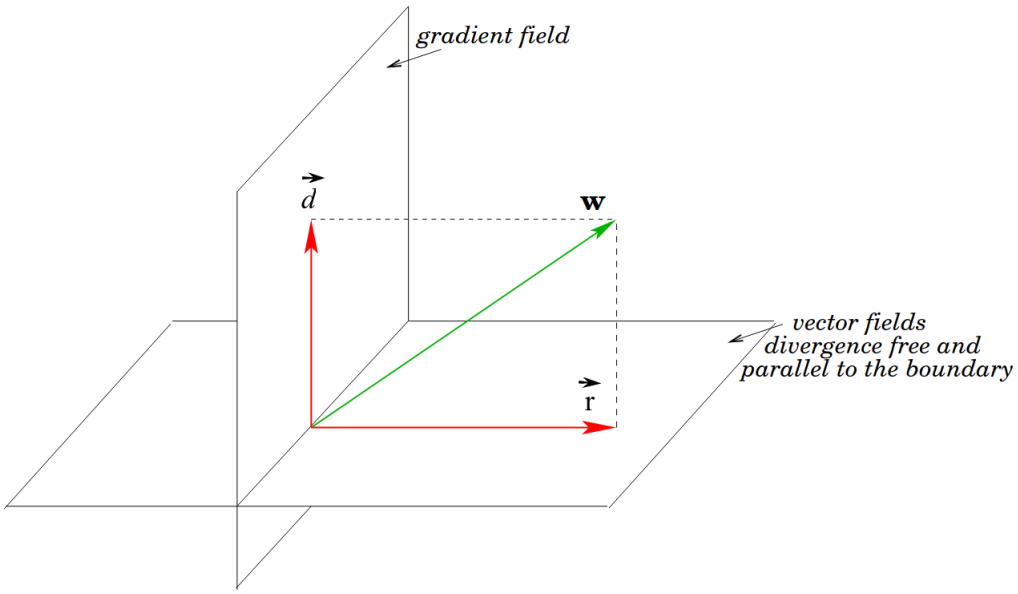


Figure 16: Vector field $\vec{\omega}$ unique decomposition [10].

4.3. Application of the HHD theorem

The Fractional Step Method (FSM) solves the incompressible Navier-Stokes equations in primitive variables by means of applying the Helmholtz-Hodge decomposition theorem [10]. Let $\Pi(\cdot)$ be a projector operator that projects a vector field over a divergence-free space parallel to the smooth boundary of its domain[7]

$$\nabla \cdot \Pi(\vec{r}) = 0. \tag{64}$$

Applying the projector operator to the Navier-Stokes equation (55)

$$\Pi\left(\frac{\partial \mathbf{u}}{\partial t} + \nabla p\right) = \Pi\left(\frac{1}{Re}\Delta \mathbf{u} - (\mathbf{u}\cdot\nabla)\mathbf{u}\right) \tag{65}$$

where the velocity field is incompressible as defined in equation (56), therefore remains unchanged when projected. On the other hand, the projection of the gradient of the pressure field turns to 0.

[7]Some basic considerations of the projector operator $\Pi(\cdot)$ are as follows:

1. An incompressible field remains unchanged under the application of the projector operator $\Pi(\cdot)$

$$\text{if } \nabla \cdot \vec{r} = 0 \longrightarrow \Pi(\vec{r}) = \vec{r} \tag{62}$$

2. The direct application to the HHD yields

$$\Pi(\vec{d}) = \Pi(\vec{\omega}) - \Pi(\vec{r}) = \vec{r} - \vec{r} = 0 \tag{63}$$

$$\Pi\left(\frac{\partial \mathbf{u}}{\partial t}\right) = \frac{\partial \mathbf{u}}{\partial t} \tag{66}$$

$$\Pi(\nabla p) = 0 \tag{67}$$

By developing equation (65) with the last statements, it is possible to split the Navier-Stokes equations in a divergence-free vector (68) and a gradient of an scalar field (69), obtaining by definition of the HHD theorem an unique decomposition.

$$\frac{\partial \mathbf{u}}{\partial t} = \Pi\left(\frac{1}{Re}\Delta\mathbf{u} - (\mathbf{u}\cdot\nabla)\mathbf{u}\right) \tag{68}$$

$$\nabla p = \frac{1}{Re}\Delta\mathbf{u} - (\mathbf{u}\cdot\nabla)\mathbf{u} - \Pi\left(\frac{1}{Re}\Delta\mathbf{u} - (\mathbf{u}\cdot\nabla)\mathbf{u}\right) \tag{69}$$

Performing the divergence of equation (69) taking into account equation (64), the *Poisson equation* for the pressure field is obtained as defined in (70).

$$\Delta p = \nabla\cdot\left(\frac{1}{Re}\Delta\mathbf{u} - (\mathbf{u}\cdot\nabla)\mathbf{u}\right) \tag{70}$$

As a conclusion, the gradient of the pressure field allows the incompressible Navier-Stokes equations to be projected onto a divergence-free space by means of the projector operator.

4.4. Numerical approach to the time integral

The time-integration scheme employed henceforth is fully explicit. For the sake of simplification, the momentum equation is rewritten, with $\boldsymbol{R}(\mathbf{u})$ standing for the convective and diffusive terms

$$\boldsymbol{R}(\mathbf{u}) = \frac{1}{Re}\Delta\mathbf{u} - (\mathbf{u}\cdot\nabla)\mathbf{u}, \tag{71}$$

therefore, equation (55) yields

$$\frac{\partial \mathbf{u}}{\partial t} = \boldsymbol{R}(\mathbf{u}) - \nabla p. \tag{72}$$

Using a central difference scheme for the transient term, a first-order backward Euler scheme for the pressure gradient term, and a fully explicit second-order Adams-Bashforth scheme for $\boldsymbol{R}(\mathbf{u})$ to integrate the momentum equation at time instant $t^{n+1/2}$, equation (72) turns to (73).

$$\frac{\mathbf{u}^{n+1} - \mathbf{u}^{n}}{\Delta t} = \frac{3}{2}\boldsymbol{R}(\mathbf{u}^{n}) - \frac{1}{2}\boldsymbol{R}(\mathbf{u}^{n-1}) - \nabla p^{n+1} \tag{73}$$

The continuity equations are implicitly integrated

$$\nabla\cdot\mathbf{u}^{n+1} = 0. \tag{74}$$

As mentioned before, these equations will be split by means of the HHD theorem. In this case, with a fictitious velocity field called predictor velocity ($\mathbf{u}^p$), which is uniquely discomposed in a divergence-free vector ($\mathbf{u}^{n+1}$) and the gradient of the pressure field (∇p^{n+1}).

$$\mathbf{u}^p = \mathbf{u}^{n+1} + \Delta t \nabla p^{n+1} \tag{75}$$

By introducing equation (75) into equation (73), an expression for the predictor velocity is obtained.

$$\mathbf{u}^p = \mathbf{u}^n + \Delta t \left(\frac{3}{2} \boldsymbol{R}(\mathbf{u}^n) - \frac{1}{2} \boldsymbol{R}(\mathbf{u}^{n-1}) \right) \tag{76}$$

Now, the divergence operator is applied to equation (75),

$$\nabla \cdot \mathbf{u}^p = \nabla \cdot \mathbf{u}^{n+1} + \Delta t \Delta p^{n+1} \tag{77}$$

yielding the Poisson equation

$$\Delta p^{n+1} = \frac{1}{\Delta t} \nabla \cdot \mathbf{u}^p. \tag{78}$$

Hence, the resolution on each time step will be based on the following algorithm [11]:

1. Evaluation of $\boldsymbol{R}(\mathbf{u}^n)$.
2. Calculation of the predictor velocity field $\mathbf{u}^p$ by means of equation (76).
3. Evaluation of $\nabla \cdot \mathbf{u}^p$ and resolution of the Poisson equation (78).
4. Calculation of the new velocity field $\mathbf{u}^{n+1}$ by using equation (75).

4.5. Time step determination

As the temporal scheme is explicit, the time step must be low enough to ensure the stability of the solver, which yields strong restrictions on the time step. The Courant-Friedrich-Levy (CFL) condition [10] stands for the maximum time step to maintain this stability. Despite this, for high Reynolds numbers, the time step must be modified by hand in order to make it short enough to obtain a converged solution on the steady state.

The CFL condition is stated as follows, where ν stands for the kinematic viscosity. The parameters C_{conv} and C_{visc} need to be small enough to ensure this stability, and therefore their values can be modified. Despite this, for the major part of the simulations the values of $C_{conv} = 0.35$ and $C_{visc} = 0.20$ will be used.

$$\Delta t_c \leq C_{conv} \left(\frac{\Delta x_i}{|u_i|} \right)_{min} \tag{79}$$

$$\Delta t_v \leq C_{visc} \left(\frac{\Delta x_i^2}{\nu} \right)_{min} \tag{80}$$

$$\Delta t = min[\Delta t_c,\ \Delta t_v] \tag{81}$$

4.6. The Checkerboard problem

From equation (75), it is noted that the computation of the new velocity needs the previous calculation of the pressure field. Isolating $\mathbf{u}^{n+1}$ and discretizing for an arbitrary node P, equation (82) is obtained.

$$u_P^{n+1} = u_P^p - \Delta t \left(\frac{p_E^{n+1} - p_W^{n+1}}{2\Delta x} \right) \tag{82}$$

Analyzing equation (82), it is possible to note that the gradient of pressure does not depend on the pressure of node P. This phenomenon can bring fully converged velocity fields with unphysical pressure distributions that verify $\nabla p^{n+1} = 0$. For instance, in the 1D spatial discretization of Figure 17, where $p_{WW}^{n+1} = 100,\ p_W^{n+1} = 0,\ p_P^{n+1} = 100,\ p_E^{n+1} = 0,$ and $p_{EE}^{n+1} = 100$, the solution would be converged with an unphysical pressure field.

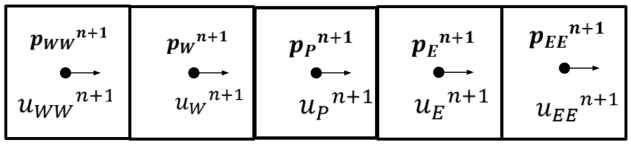


Figure 17: Example of the Checkerboard problem [12].

There are two possible solutions to this problem: the use of staggered or collocated meshes. Henceforth, the focus will be set on the staggered meshes.

4.7. Staggered meshes

The staggered velocity mesh solves the Checkerboard problem. Despite being relatively complex to implement it over unstructured meshes, the implementation of structured staggered meshes is widely used for academic purposes due to its relative simplicity. In a two-dimensional problem, there exist the main mesh (where the pressure field is defined), the staggered-x mesh and the staggered-y mesh (where the velocity fields on x and y directions are defined, respectively). A schematic view of these meshes is shown in Figure 18.

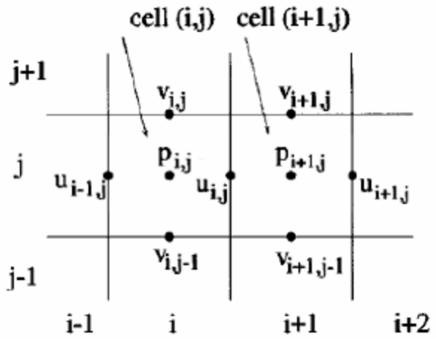


Figure 18: Staggered mesh illustration [13].

4.8. Numerical discretization

The fist step of the Fractional Step Method is to obtain $\boldsymbol{R}(\mathbf{u})$. In order to discretize this variable in the x-direction, it is integrated over an arbitrary staggered-x control volume (Ω_x) and the Gauss's divergence theorem is applied.

$$\int_{\Omega_x} R(u)d\Omega_x = \int_{\Omega_x} \frac{1}{Re}\Delta u d\Omega_x - \int_{\Omega_x} (\mathbf{u}\cdot\nabla)u d\Omega_x = \\ = \int_{\partial\Omega_x} \frac{1}{Re}\nabla u\cdot\mathbf{n}dS - \int_{\partial\Omega_x}(\mathbf{u})u\cdot\mathbf{n}dS \tag{83}$$

Now, discretizing equation (83)

$$R(u)\Omega_{xP} \approx -\frac{1}{\rho}\left[\dot{m}_e u_e - \dot{m}_w u_w + \dot{m}_n u_n - \dot{m}_s u_s\right] + \\ \frac{1}{Re}\left[\frac{u_E - u_P}{d_{EP}}A_e - \frac{u_P - u_W}{d_{WP}}A_w + \frac{u_N - u_P}{d_{NP}}A_n - \frac{u_P - u_S}{d_{SP}}A_s\right] \tag{84}$$

where $\dot{m}_e = (\rho u)_e A_e$, $\dot{m}_w = (\rho u)_w A_w$, $\dot{m}_n = (\rho v)_n A_n$, and $\dot{m}_s = (\rho v)_s A_s$. In regard to the discretization of $\boldsymbol{R}(\mathbf{u})$ in the y-direction, the process is analogous, yielding equation (85).

$$R(v)\Omega_{yP} \approx -\frac{1}{\rho}\left[\dot{m}_e v_e - \dot{m}_w v_w + \dot{m}_n v_n - \dot{m}_s v_s\right] + \\ \frac{1}{Re}\left[\frac{v_E - v_P}{d_{EP}}A_e - \frac{v_P - v_W}{d_{WP}}A_w + \frac{v_N - v_P}{d_{NP}}A_n - \frac{v_P - v_S}{d_{SP}}A_s\right] \tag{85}$$

The velocity field evaluation on the interfaces of the control volumes is performed by means of convective numerical schemes, some of them described on the Smith-Hutton Section 3.5. For the sake of simplification, during the development of the FSM for the Driven Cavity problem, only the Central Difference Scheme (CDS) and the Quadratic Upstream Interpolation for Convective Kinematic (QUICK) scheme are programmed.

As the QUICK scheme has not been introduced yet, a brief introduction is given. It consists on a high order scheme that needs three nodes for the interpolation: the upstream (U), central (C) and downstream (D) nodes. Depending of the direction of the volumetric flow on the interface of interest, these three nodes will have different positions. For instance, if the intention is to calculate an arbitrary variable ϕ on the east interface (ϕ_e) and the volumetric flow is positive, the upstream node will be W, the central node P, and the downstream node E. Otherwise, if the volumetric flow is negative, the upstream node will be EE, the central E, and the downstream P. Two dimensionless variables are defined, regarding the variable ϕ and the position of the nodes as shown in equation (86)

$$\hat{\phi} = \frac{\phi - \phi_U}{\phi_D - \phi_U} \qquad \hat{x} = \frac{x - x_U}{x_D - x_U} \tag{86}$$

and the dimensionless variable at the east interface is computed with equation (87).

$$\hat{\phi_e} = \hat{x}_e + \frac{\hat{x}_e(\hat{x}_e - 1)}{\hat{x}_C(\hat{x}_C - 1)}(\hat{\phi}_C - \hat{x}_C) \tag{87}$$

Once $\boldsymbol{R}(\mathbf{u})$ is obtained, the predictor velocity field is computed in the staggered meshes by means of equation (76), obtaining equations (88) and (89).

$$u^p = u^n + \Delta t \left(\frac{3}{2} R(u^n) - \frac{1}{2} R(u^{n-1}) \right) \tag{88}$$

$$v^p = v^n + \Delta t \left(\frac{3}{2} R(v^n) - \frac{1}{2} R(v^{n-1}) \right) \tag{89}$$

The next step for the numerical resolution of the FSM is to solve the Poisson equation. In order to do so, equation (78) will be integrated over an arbitrary control volume within the main mesh (Ω) and the Gauss theorem applied.

$$\begin{aligned} \int_\Omega \Delta p^{n+1} d\Omega &= \frac{1}{\Delta t} \int_\Omega \nabla \cdot \mathbf{u}^p d\Omega \\ \int_{\partial\Omega} \nabla p^{n+1} \cdot \mathbf{n} dS &= \frac{1}{\Delta t} \int_{\partial\Omega} \mathbf{u}^p \cdot \mathbf{n} dS \end{aligned} \tag{90}$$

Discretizing equation (90), it is possible to obtain:

$$\begin{aligned} \frac{p_E^{n+1} - p_P^{n+1}}{d_{EP}} A_e - \frac{p_P^{n+1} - p_W^{n+1}}{d_{WP}} A_w + \frac{p_N^{n+1} - p_P^{n+1}}{d_{NP}} A_n - \frac{p_P^{n+1} - p_S^{n+1}}{d_{SP}} A_s = \\ \frac{1}{\Delta t} \left[u_e^p A_e - u_w^p A_w + v_n^p A_n - v_s^p A_s \right] \end{aligned} \tag{91}$$

A linear solver is needed for the resolution of equation (91). In the case of the present dissertation, a Gauss-Seidel algorithm with relaxation factor. Rearranging the terms,

$$a_P p_P^{n+1} = a_E p_E^{n+1} + a_W p_W^{n+1} + a_N p_N^{n+1} + a_S p_S^{n+1} + b_P \tag{92}$$

and the coefficients correspond to:

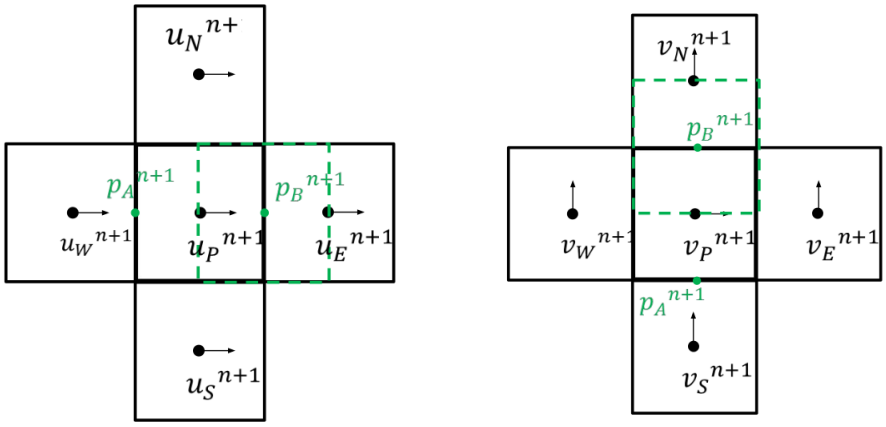
$$\begin{cases} a_E = \frac{A_e}{d_{EP}} \qquad a_W = \frac{A_w}{d_{WP}} \\ a_N = \frac{A_n}{d_{NP}} \qquad a_S = \frac{A_s}{d_{SP}} \\ a_P = a_E + a_W + a_N + a_S \\ b_P = -\frac{1}{\Delta t} \left[u_e^p A_e - u_w^p A_w + v_n^p A_n - v_s^p A_s \right]. \end{cases} \tag{93}$$

Note that a pressure boundary condition $\partial P/\partial n$ over a wall would be reflected as all the coefficients being 0 but a_p and the normal to the wall, which would be equal to 1. The last step of the FSM stands for the calculation of the new velocity field $\mathbf{u}^{n+1}$. Thanks to the implementation of the staggered meshes, the gradient of the pressure field over an arbitrary node P will depend on the pressure at the same node, avoiding the Checkerboard problem. Therefore, discretizing equation (75) in the two dimensions,

$$u_P^{n+1} = u_P^p - \Delta t \frac{p_B^{n+1} - p_A^{n+1}}{d_{BA}}, \tag{94}$$

$$v_P^{n+1} = v_P^p - \Delta t \frac{p_B^{n+1} - p_A^{n+1}}{d_{BA}}, \tag{95}$$

where an schematic view of these meshes is shown in Figure 19.



(a) Horizontal velocity in the staggered-x mesh.

(b) Vertical velocity in the staggered-y mesh.

Figure 19: Calculation of the new velocity field by means of staggered meshes [12].

5. The Lid Driven Cavity Problem

5.1. Problem definition

The Lid Driven Cavity consists on a two-dimensional box of dimensions $L \times L$ with a fluid within, which has a dynamic viscosity μ and density ρ, as shown in Figure 20. All the walls are static $u = 0\ m/s$ and $v = 0\ m/s$ but the top one, which has an horizontal constant velocity of $u(x, y = L) = U_{ref}$ and $v = 0\ m/s$. The pressure boundary condition is $\partial P/\partial n = 0$ for the four walls.

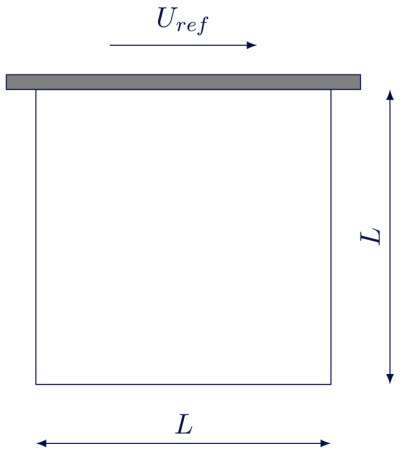


Figure 20: Illustration of the Lid Driven Cavity geometry.

The Reynolds number is characterized by $Re = \frac{\rho U_{ref} L}{\mu}$, where $U_{ref} = 1\ m/s$ and $L = 1\ m$. The objective is to develop a C++ code able to find the velocity in the steady state for different Reynolds numbers within a range from $Re = 100$ to $Re = 10.000$.

5.2. Algorithm

A C++ algorithm is programmed in order to find the steady solution by means of the transient evolution of the system. The main functional blocks of the mentioned algorithm are defined below.

1. **Input data.**
   a) Definition of the problem parameters: geometric limits of the domain (L), velocity of the top wall (U_{ref}) and Reynolds number (Re).
   b) Initial pressure and velocity fields for the first iteration: P_{ini}, u_{ini} and v_{ini}. These values are set to 0 for the sake of simplification.
   c) Numerical parameters for the meshes discretization: N for the x-direction and M for the y-direction. The main mesh is $N \times M$.
   d) Numerical parameters for the solver: δ for the Gauss-Seidel tolerance, fr for the relaxation factor, tol for the transient algorithm convergence criterion, $delta\_t$ for the time step of the first iteration, t_{max} for the maximum simulation time in case of not finding a solution, $SCHEME$ for the selection of the convective interpolation scheme (0 stands for CDS and 1 stands for QUICK), Q in case of manually modifying the time step ($delta\_t/Q$).

2. **Previous calculus.**
   a) Generation of the structured main mesh, staggered-x and staggered-y meshes: definition of the geometry and assignation of the pressure and velocity fields to P_{ini}, u_{ini} and v_{ini}. The top wall condition U_{ref} is taken into account in the generation of the staggered-x mesh. The main mesh for the pressure field is defined at t^{n+1}, with an auxiliary field for the assignation of p^* in the Gauss-Seidel solver. The velocity fields are defined at the instant t^{n-1}, t^n, t^{n+1}, initializing the predictor velocity field for both staggered meshes.
   b) Initialization of the discretization coefficients for the Poisson equation Gauss-Seidel solver, where all the coefficients start at 0.
   c) Initialization of time variable at $t = 0$ *sec*.
3. **Main loop.**
   a) Predictor velocity calculation: a function for the calculation of $\boldsymbol{R}(\mathbf{u})$ is included, which takes into account the chosen interpolation scheme. Equations (88) and (89) are solved.
   b) The coefficients for the Poisson Gauss-Seidel are defined by computing equation (93). The pressure boundary conditions are taken into account on the walls.
   c) Two-dimensional Gauss-Seidel algorithm for the pressure field.
      i. Apply equation (92) from the left to right, from the bottom to the top side nodes to calculate p^{n+1}, using p^* when the value of the scalar field is unknown and the last calculated value of p^{n+1} when possible. The relaxation factor is taken into account.
      ii. Is $max|p^{n+1} - p^*| < \delta$?
         A. If not, $p^* = p^{n+1}$ and go to (3ci).
         B. If yes, go to (3d).
   d) Calculation of the new velocity field by means of computing equations (94) and (95). The velocity boundary conditions on the walls are imposed on the respective nodes.
   e) Computation of the new time step by considering the CFL condition. The calculated time step is divided by the input parameter Q, as the user is able to modify it by hand in order to ensure the stability of the solver.
   f) Stop calculations and convergence criterion. Is $t < t_{max}$?
      i. If not, the solution is not converged within the specified maximum time. The loop is stopped and an error message is displayed on the screen.
      ii. If yes, is $max|u^{n+1} - u^n|$ AND $max|v^{n+1} - v^n| < \ tol$?
         A. If not, $t = t + \Delta t$, $u^{n-1} = u^n$, $u^n = u^{n+1}$, $v^{n-1} = v^n$ and $v^n = v^{n+1}$. Go to (3a).
         B. If yes, the solution is converged and the loop stopped. A message is displayed on the screed to notify it. Go to (4).
4. **Post-process.**
   a) A function generates output files with the geometry of the meshes, results of the simulation (pressure and velocity fields), and a list with the simulation data (input parameters and time until the converged solution).
   b) Plotting of the evolution of the velocity fields at $L/2$, and comparison of the results with the reference values for different Reynolds numbers and mesh size. Plotting of the streamlines.

Finally, Figure 21 is a representation of the above explained algorithm in the form of a flux diagram.

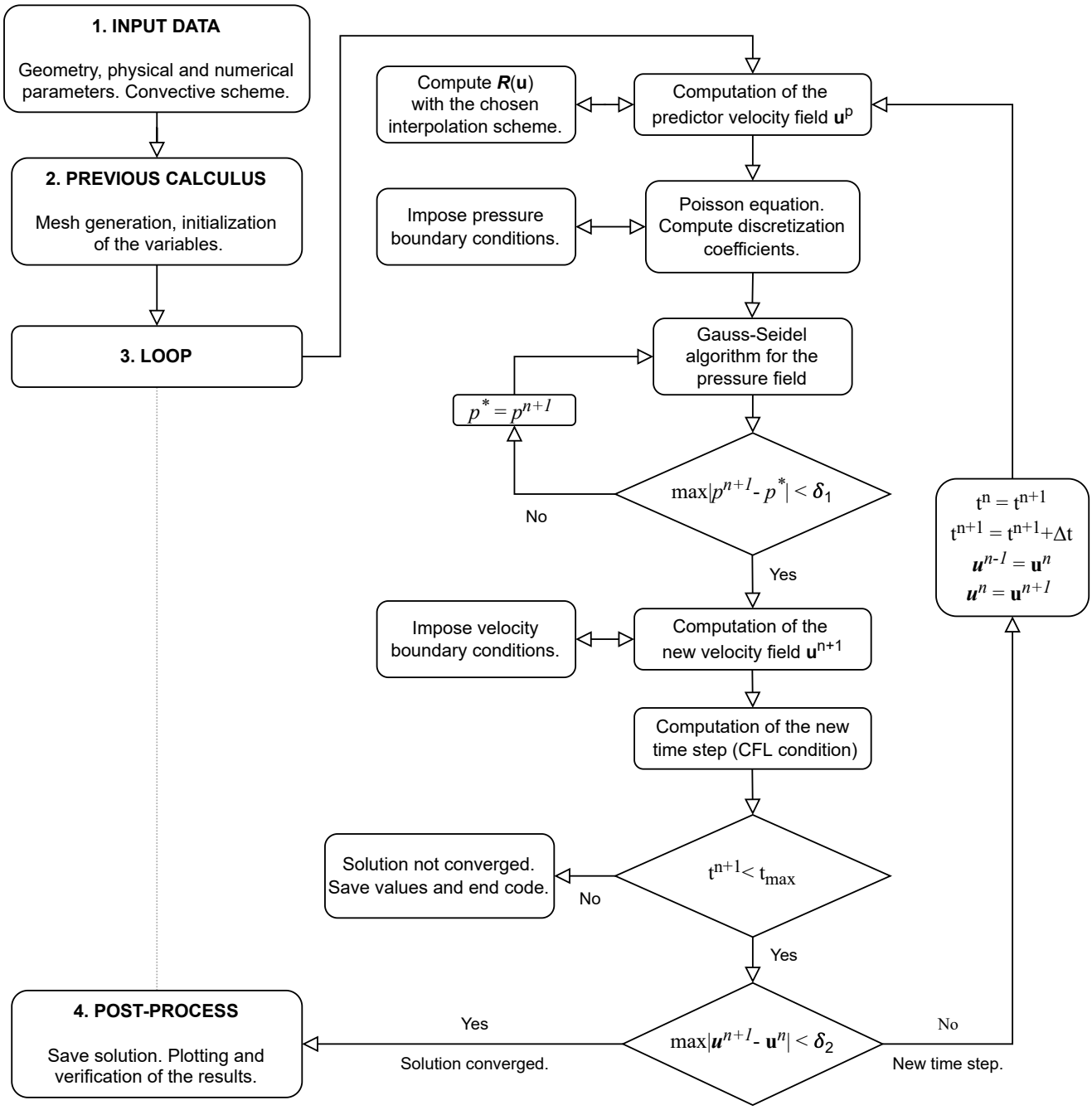


Figure 21: Lid Driven Cavity. Resolution algorithm.

5.3. Results

The simulation is carried out for different Reynolds numbers: 100, 400, 1000, 3200, 5000, 7500 and 10.000. In order to evaluate the correct convergence of the results, different mesh sizes have been considered and compared with actual data extracted from references. Furthermore, streamline plots with the intensity of the velocity field are also included, taking into account the modulus of the velocity. Note that for the sake of simplification, all the simulations have been carried out by means of a CDS convective interpolation scheme.

On the one hand, Tables 8 and 9 deliver the obtained velocity at the positions of the reference data for $x = L/2$ and $y = L/2$, respectively. On the other hand, Figure 22 shows the horizontal and vertical velocity distributions for different Reynolds numbers and mesh sizes at $x = L/2$ and $y = L/2$ compared with data extracted from references. Finally, Figure 23 stands for the intensity of the velocity field and its streamlines for the different Reynolds numbers and a mesh size of 128×128 elements.

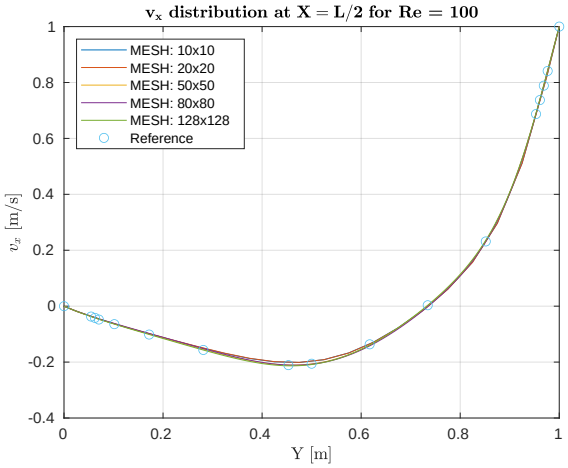
Regarding the tables with the data obtained from the simulations, they can be compared with the reference data from Tables A.2 and A.3 of the Appendix, extracted from [14].

y/L	$Re = 100$	$Re = 400$	$Re = 1000$	$Re = 3200$	$Re = 5000$	$Re = 7500$	$Re = 10000$
1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
0.9766	0.8433	0.7594	0.6604	0.5318	0.4706	0.4390	0.4475
0.9688	0.7913	0.6857	0.5762	0.4653	0.4450	0.4315	0.4541
0.9609	0.7404	0.6191	0.5123	0.4422	0.4419	0.4332	0.4494
0.9531	0.6908	0.5605	0.4669	0.4368	0.4420	0.4318	0.4467
0.8516	0.2357	0.2900	0.3307	0.3273	0.3193	0.2927	0.3160
0.7344	0.0033	0.1615	0.1849	0.1833	0.1827	0.1592	0.1951
0.6172	-0.1391	0.0206	0.0556	0.0710	0.0708	0.0603	0.0839
0.5000	-0.2087	-0.1153	-0.0612	-0.0348	-0.0287	-0.0213	-0.0195
0.4531	-0.2133	-0.1716	-0.1065	-0.0753	-0.0675	-0.0530	-0.0610
0.2813	-0.1568	-0.3263	-0.2766	-0.2308	-0.2133	-0.1833	-0.2126
0.1719	-0.1012	-0.2413	-0.3811	-0.3298	-0.3113	-0.2828	-0.3021
0.1016	-0.0641	-0.1448	-0.2939	-0.4107	-0.3902	-0.3554	-0.3629
0.0703	-0.0464	-0.1024	-0.2186	-0.3842	-0.4107	-0.4006	-0.4059
0.0625	-0.0418	-0.0918	-0.1986	-0.3615	-0.4003	-0.4039	-0.4056
0.0547	-0.0370	-0.0811	-0.1780	-0.3330	-0.3813	-0.3981	-0.4052
0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000

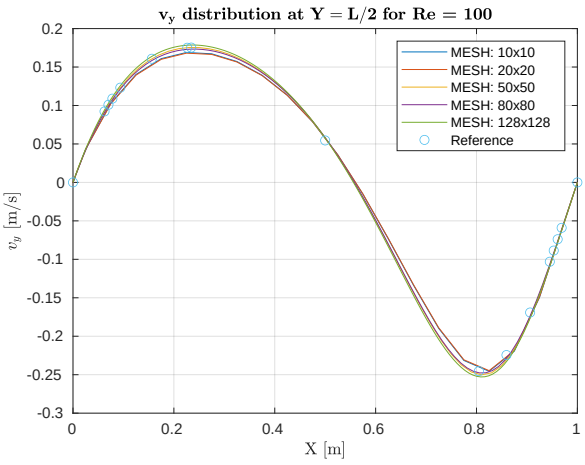
Table 8: Lid Driven Cavity. Horizontal velocity distribution at $x = L/2$ for different Re numbers, in m/s.

x/L	$Re = 100$	$Re = 400$	$Re = 1000$	$Re = 3200$	$Re = 5000$	$Re = 7500$	$Re = 10000$
0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
0.0625	0.0942	0.1831	0.2744	0.3729	0.3999	0.4047	0.4143
0.0703	0.1029	0.1966	0.2897	0.3882	0.4096	0.4069	0.4145
0.0781	0.1110	0.2087	0.3032	0.3995	0.4142	0.4037	0.4070
0.0938	0.1256	0.2292	0.3258	0.4031	0.4106	0.3874	0.3951
0.1563	0.1638	0.2808	0.3696	0.3666	0.3388	0.3088	0.3290
0.2266	0.1784	0.3014	0.3291	0.2783	0.2645	0.2333	0.2717
0.2344	0.1786	0.3011	0.3207	0.2696	0.2565	0.2254	0.2604
0.5000	0.0576	0.0526	0.0258	0.0136	0.0100	0.0066	0.0087
0.8047	-0.2525	-0.3836	-0.3155	-0.2968	-0.2843	-0.2560	-0.2877
0.8594	-0.2329	-0.4508	-0.4215	-0.3587	-0.3480	-0.3208	-0.3541
0.9063	-0.1767	-0.2475	-0.5171	-0.4226	-0.4018	-0.3772	-0.4002
0.9453	-0.1082	-0.2332	-0.4027	-0.5383	-0.5096	-0.4623	-0.4597
0.9531	-0.0931	-0.1968	-0.3483	-0.5269	-0.5285	-0.4929	-0.4851
0.9609	-0.0777	-0.1603	-0.2874	-0.4782	-0.5216	-0.5108	-0.4988
0.9688	-0.0621	-0.1244	-0.2232	-0.3870	-0.4708	-0.4915	-0.5056
1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000

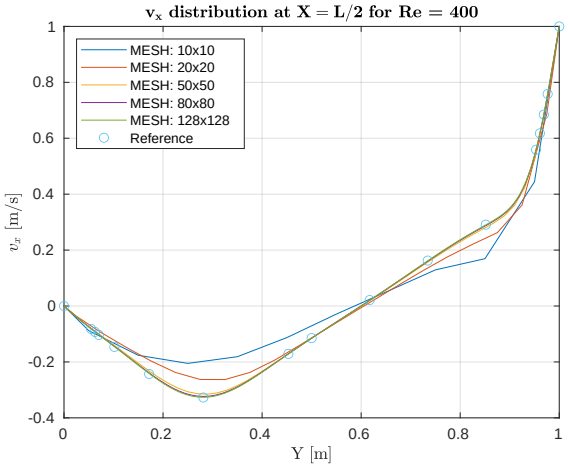
Table 9: Lid Driven Cavity. Vertical velocity distribution at $y = L/2$ for different Re numbers, in m/s.



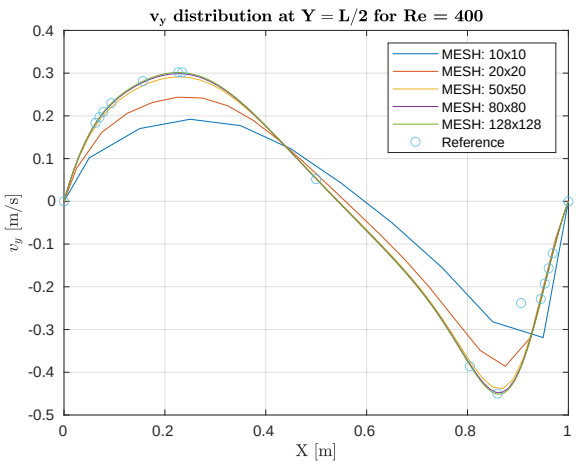
(a) Horizontal velocity for $Re = 100$.



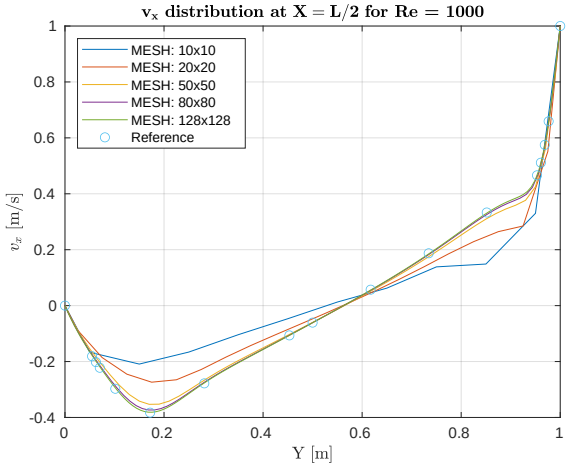
(b) Vertical velocity for $Re = 100$.



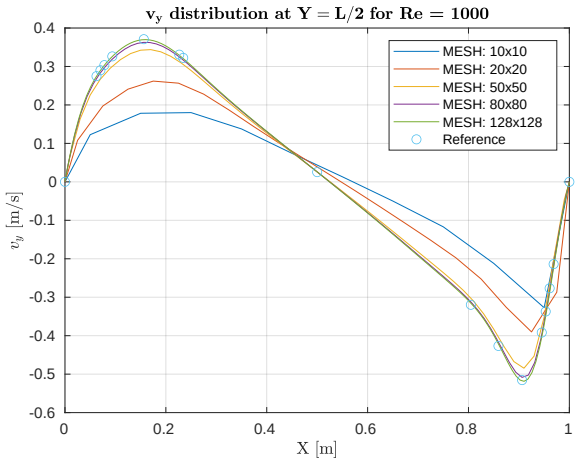
(c) Horizontal velocity for $Re = 400$.



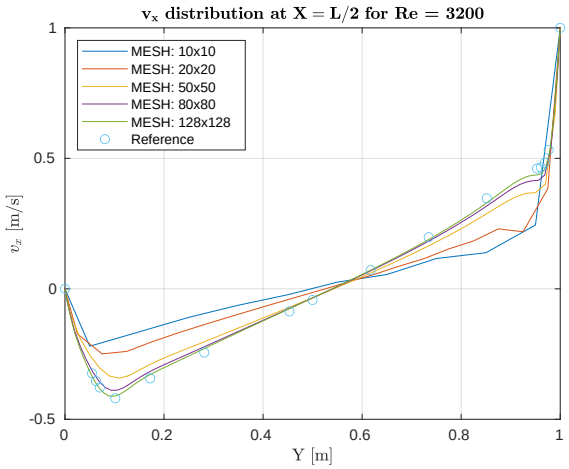
(d) Vertical velocity for $Re = 400$.



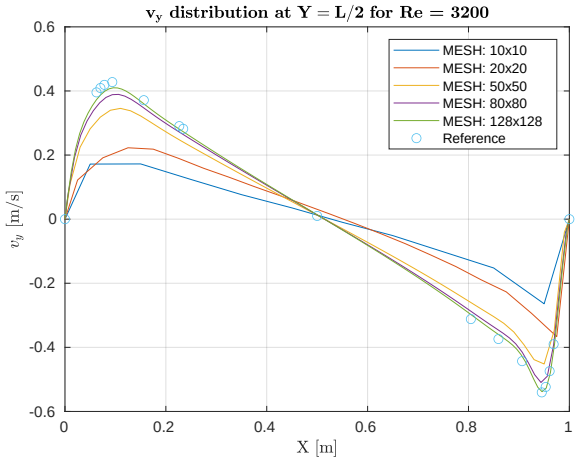
(e) Horizontal velocity for $Re = 1000$.



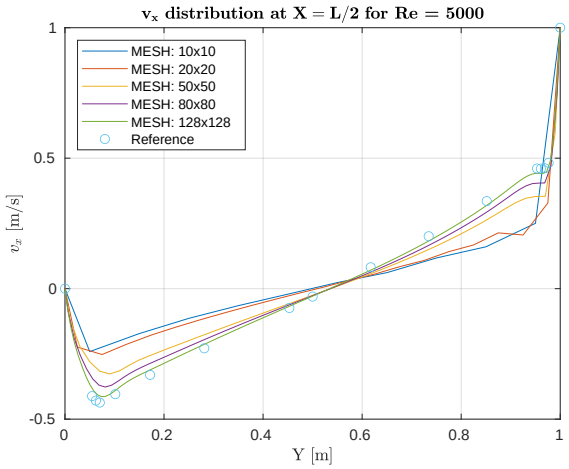
(f) Vertical velocity for $Re = 1000$.



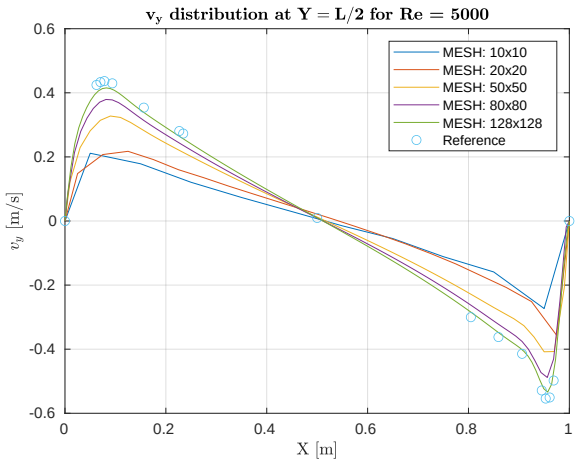
(g) Horizontal velocity for $Re = 3200$.



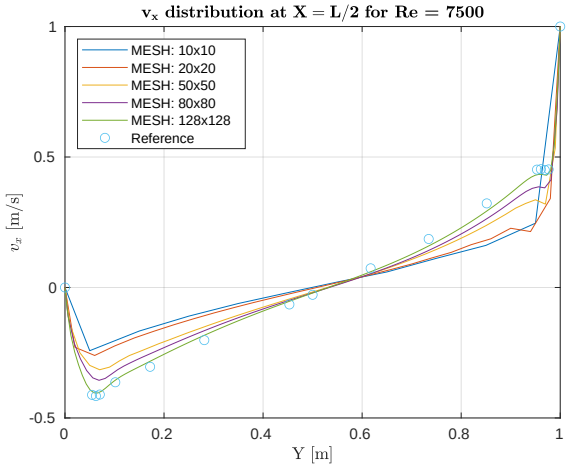
(h) Vertical velocity for $Re = 3200$.



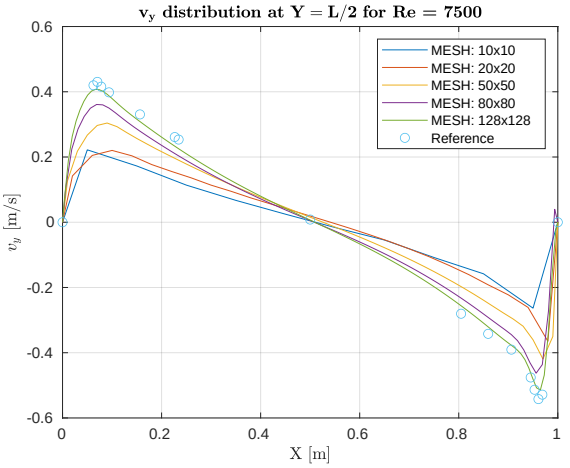
(i) Horizontal velocity for $Re = 5000$.



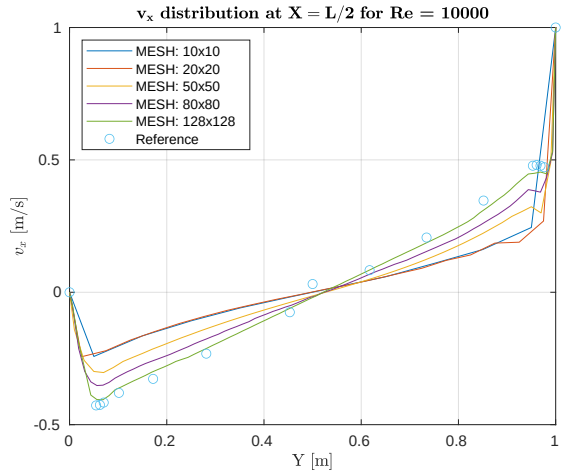
(j) Vertical velocity for $Re = 5000$.



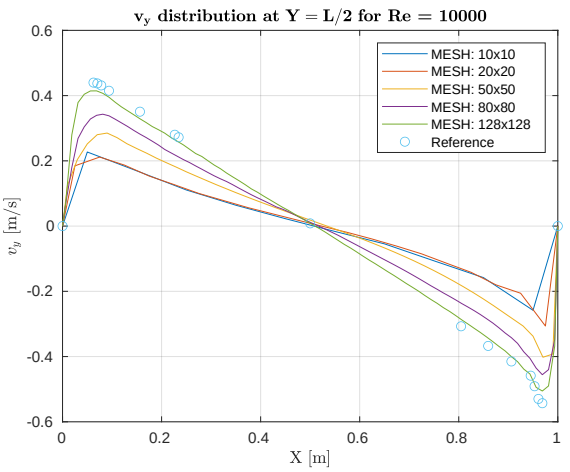
(k) Horizontal velocity for $Re = 7500$.



(l) Vertical velocity for $Re = 7500$.



(m) Horizontal velocity for $Re = 10000$.



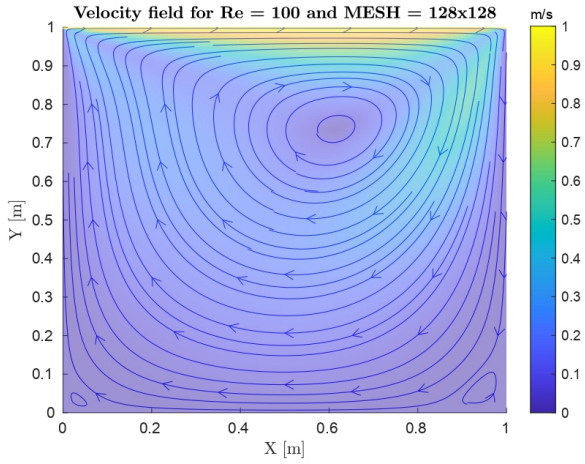
(n) Vertical velocity for $Re = 10000$.

Figure 22: Lid Driven Cavity. Velocity distributions compared with reference data.

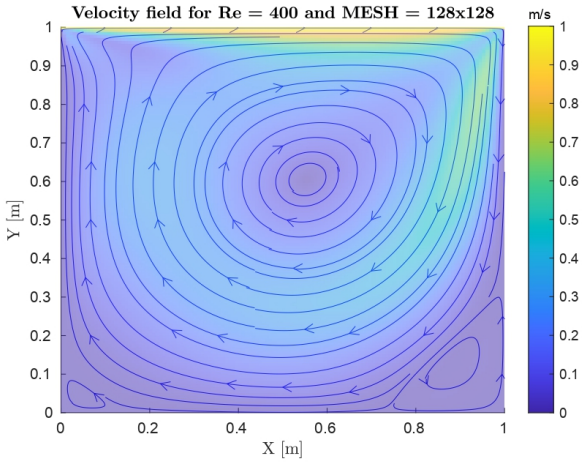
The horizontal velocity at $x = L/2$ starts always in $u = 0\ m/s$ and ends in $u = 1\ m/s$, as expected due to the boundary conditions of the problem. For low Reynolds numbers ($Re = 100$), the distribution is smooth with a negative minimum that tends to be placed in $Y = L/2$. As the Reynolds number increases, the velocity variations near the walls become more intense, rapidly decreasing in the bottom wall and increasing in the top wall. For high Reynolds numbers, the sudden variation in the walls is considerable and the distribution gets to be a positive straight line in most of the y-direction.

The vertical velocity at $y = L/2$ always starts and ends in $v = 0$, due to the boundary conditions of the problem. For low Reynolds numbers, the distribution is smooth with a clear maximum and a minimum located in the first and second half of the x dimension, respectively. The absolute value of the intensity of the minimum is higher than the one of the maximum. As the Reynolds increases, the variation near to the walls gets exaggerated, and the distribution tends to be a negative straight line in most of the x-direction.

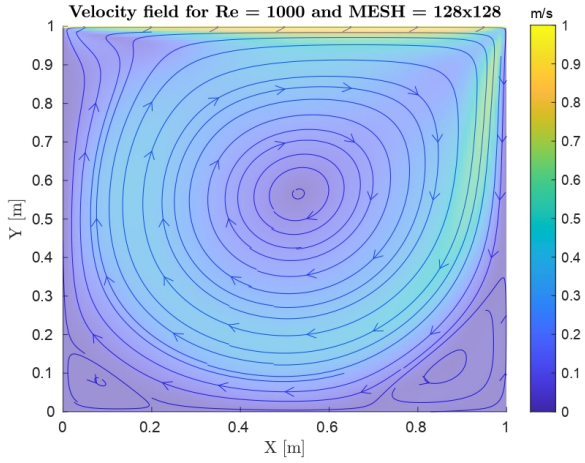
For both cases, the results obtained in the simulations are very close to the ones extracted from reference [14]. The differences start to be considerable when the Reynolds number is higher than 3200, as the fluid starts to be affected by turbulent effects. This yields to an increase of the computation time, as the convergence for $Re = 10000$ needs many more iterations than for $Re = 100$. Moreover, the density of the mesh needs to be increased to obtained fully converged solutions.



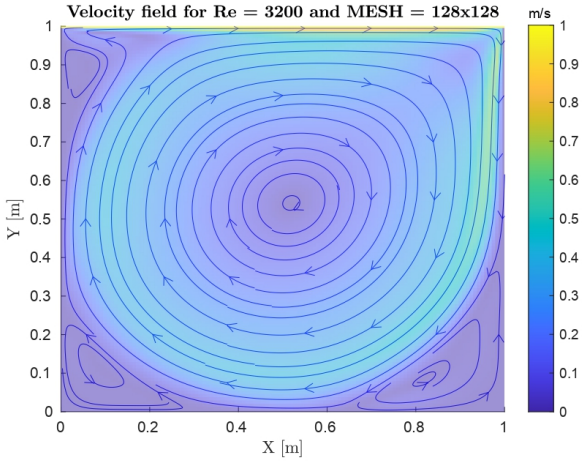
(a) $Re = 100$.



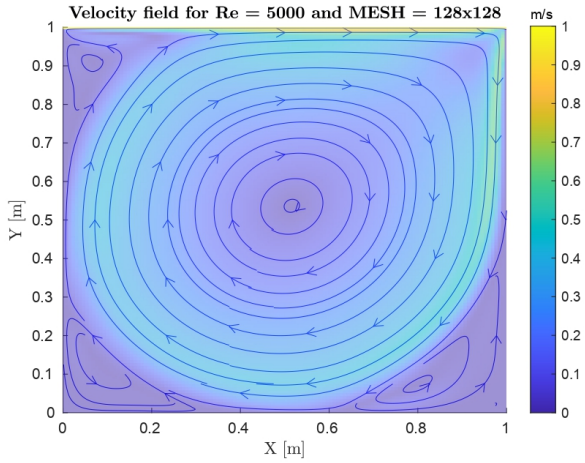
(b) $Re = 400$.



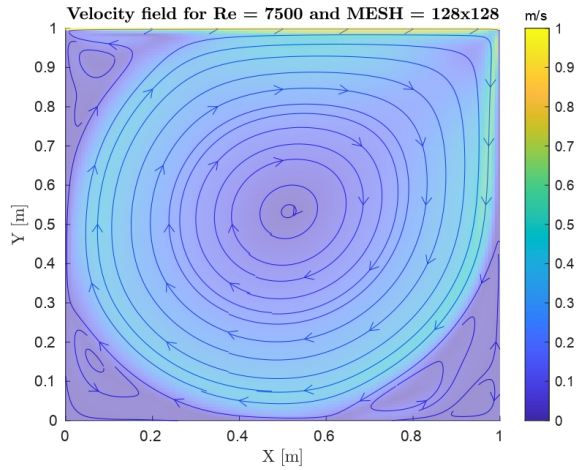
(c) $Re = 1000$.



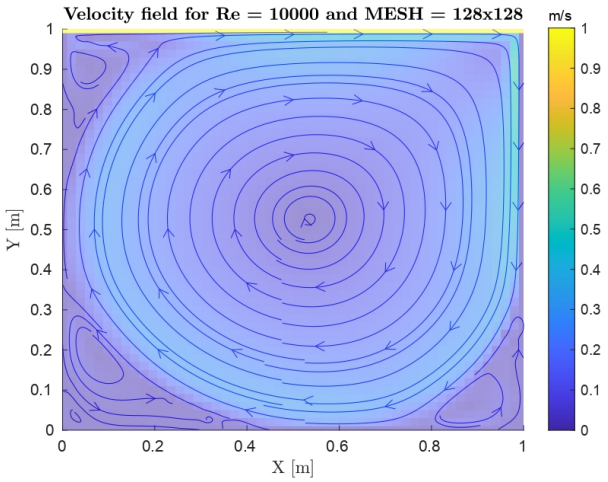
(d) $Re = 3200$.



(e) $Re = 5000$.



(f) $Re = 7500$.



(g) $Re = 10000$.

Figure 23: Lid Driven Cavity. Velocity field intensity and streamlines for a 128x128 mesh.

Regarding the velocity field intensity and streamlines plots, for low Reynolds numbers ($Re = 100$), a vortex appears nearby the centre of the domain. The flow moves in clockwise direction, as the top wall shifts from left to right and induces movement to the fluid. The central vortex starts at an upper right position and tends to move to the centre of the domain when the Reynolds increases. Two more vortexes are created at the bottom corners, which magnitude increases with the Reynolds number. A fourth vortex is induced in the upper left corner for $Re > 3200$. For high Reynolds numbers ($Re = 7500$ and $Re = 10000$), the bottom vortexes start to be affected by turbulence effects, creating recirculation zones with non-steady motion.

6. The Differentially Heated Cavity Problem

6.1. Problem definition

The objective of the Differentially Heated Cavity problem is to obtain the velocity, temperature and pressure fields taking into account the effect of a natural convection. A fluid is enclosed in a two-dimensional cavity of dimensions $L \times L$ with a hot left wall ($T_{hot} = 1\ K$) and a cold right wall ($T_{cold} = 0\ K$).

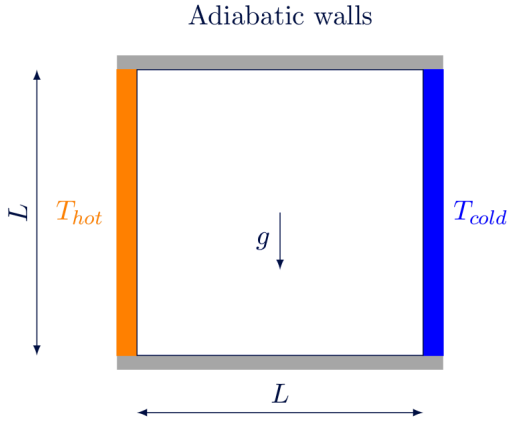


Figure 24: Illustration of the differentially heated cavity geometry.

All the walls are static ($u = 0\ m/s$ and $v = 0\ m/s$), and the pressure boundary condition is $\partial P/\partial n = 0$. The top and bottom walls are adiabatic. All the thermophysical properties of the fluid are known.

6.2. Theoretical background

6.2.1. Dimensionless Navier-Stokes equations

The continuity and momentum Navier-Stokes equations are defined by (96) and (97). The main difference regarding the Lid Driven Cavity problem is the addition of the Boussinesq term in the momentum equation, which allows to take into consideration a natural convection situation. Moreover, equation (98) is the convection-diffusion equation included for the resolution of the temperature field.

$$\nabla \cdot \mathbf{u} = 0 \tag{96}$$

$$\rho \frac{\partial \mathbf{u}}{\partial t} + \rho(\mathbf{u} \cdot \nabla)\mathbf{u} = -\nabla p + \mu \Delta \mathbf{u} + \rho \mathbf{g} \beta \left(T - T_{\infty}\right) \tag{97}$$

$$\frac{\partial T}{\partial t} + \mathbf{u} \cdot \nabla T = \frac{\lambda}{\rho c_p} \Delta T \tag{98}$$

As usual, ρ and μ are the density and viscosity of the fluid, β is the coefficient of volumetric expansion, $\mathbf{g}$ is the gravity, $\frac{\lambda}{\rho c_p}$ is the thermal diffusivity α. Now, the following dimensionless variables are defined:

$$x^* = \frac{x}{L}, \qquad y^* = \frac{y}{L}, \qquad T^* = \frac{T - T_{cold}}{T_{hot} - T_{cold}}, \qquad u^* = \frac{uL}{\alpha}, \qquad v^* = \frac{vL}{\alpha}, \qquad p^* = \frac{p}{p_0}.$$

Introducing these definitions into the Navier-Stokes equations, it is possible to obtain equations (99), (100) and (101),

$$\nabla \cdot \mathbf{u}^* = 0 \tag{99}$$

$$\frac{\partial \mathbf{u}^*}{\partial t} + (\mathbf{u}^* \cdot \nabla)\mathbf{u}^* = -\nabla p^* + \frac{1}{Re}\Delta \mathbf{u}^* + RaPrT^* \tag{100}$$

$$\frac{\partial T^*}{\partial t} + \mathbf{u}^* \cdot \nabla T^* = \Delta T^* \tag{101}$$

where the Reynolds[8], Prandtl and Rayleigh[9] numbers are defined below:

$$Re = \frac{1}{Pr}, \qquad Pr = \frac{\mu c_p}{\lambda}, \qquad Ra = \frac{c_p \mathbf{g} \beta \rho^2 \left(T_{hot} - T_{cold}\right) L^3}{\mu \lambda}. \tag{102}$$

Going back to equation (71), the dimensionless momentum equation is rewritten by means of $\boldsymbol{R}(\mathbf{u}^*)$, which now takes the following form:

$$\boldsymbol{R}(\mathbf{u}^*) = \frac{1}{Re}\Delta \mathbf{u}^* - (\mathbf{u}^* \cdot \nabla)\mathbf{u}^* + RaPrT^*. \tag{103}$$

As can be seen, the Boussinesq term is added and the equation is expressed in dimensionless variables. Thus, the algorithm for the resolution of the velocity field will be the same as already defined, with the only difference being the discretization in the y-direction of $\boldsymbol{R}(\mathbf{u}^*)$, which is defined in equation (104).

$$R(v^*)\Omega_{yP} \approx -\frac{1}{\rho}\left[\dot{m}_e v_e^* - \dot{m}_w v_w^* + \dot{m}_n v_n^* - \dot{m}_s v_s^*\right] + \tag{104}$$
$$\frac{1}{Re}\left[\frac{v_E^* - v_P^*}{d_{EP}}A_e - \frac{v_P^* - v_W^*}{d_{WP}}A_w + \frac{v_N^* - v_P^*}{d_{NP}}A_n - \frac{v_P^* - v_S^*}{d_{SP}}A_s\right] + RaPrT^*\Omega_{yP}$$

Regarding the resolution of the convection-diffusion equation (101), it is solved by exactly the same discretization equation as for the Smith-Hutton problem, defined in (50). The programmed convective interpolation schemes are also the same (CDS, UDS, HDS, PLDS and EDS) and the resolution algorithm is completely analogous. The only details to take into consideration is that the parameter Γ takes the value of $\Gamma = 1$, and the application of the boundary conditions.

In regard to the adiabatic walls boundary conditions, they appear in the calculus of the convection-diffusion equation discretization coefficients, as all the coefficients become 0 but a_p and the normal to the wall, which are equal to 1. On the other hand, the isotherm walls boundary conditions make all the coefficients 0 but a_p and b_p, which are equal to 1 and the temperature of the wall, respectively.

6.2.2. Calculation of the Nusselt number

The Nusselt number will be calculated by integrating the heat flux over a vertical line at a certain position x. The heat flux in the x-direction in dimensionless variables is written as:

$$\dot{q}_x\left(x^*, y^*\right) = u^* T^* - \frac{\partial T^*}{\partial x^*}. \tag{105}$$

[8]Note that the reference velocity is $\frac{\alpha}{L}$. Therefore, $Re = \frac{\rho\alpha}{\mu} = \frac{\rho\lambda}{\rho c_p \mu}$ and the relation $Re = \frac{1}{Pr}$ is obtained.
[9]The Rayleigh number will only exist in the vertical direction, as there is no gravity in the horizontal dimension.

Moreover, the overall heat flux along a vertical line (Nu_x) is:

$$Nu_x = \int_0^1 \dot{q}_x\left(x^*, y^*\right) dy^* = \int_0^1 \left(u^* T^* - \frac{\partial T^*}{\partial x^*}\right) dy^*. \tag{106}$$

The heat flux should be constant and independent of the vertical line of integration. Despite this, some errors can be induced due to the numerical discretization. Equation (106) is discretized to obtain equation (107), where u_P^* is calculated by interpolating the horizontal velocity in the staggered-x mesh.

$$Nu_x = \frac{1}{y_f^* - y_i^*} \sum_j \left(u_P^* T_P^* - \frac{T_E^* + T_P^*}{2d_{EP}} + \frac{T_P^* + T_W^*}{2d_{WP}}\right) \Delta y_j^* \tag{107}$$

The parameters y_i^* and y_f^* are the initial and final positions within the integration (0 and L), and the subscript j indicates the number of the element in the vertical direction. Finally, the average Nusselt number ($\overline{Nu}$) is obtained by means of equation (108), which can be discretized to obtain equation (109). The parameters x_i^*, x_f^* and the subscript i are analogous to the ones mentioned above, but in the horizontal direction.

$$\overline{Nu} = \int_0^1 Nu_x dx^* \tag{108}$$

$$\overline{Nu} = \frac{1}{x_f^* - x_i^*} \sum_i Nu_x \Delta x_i^* \tag{109}$$

6.3. Algorithm

The programmed C++ algorithm is summarized below.

1. **Input data.**
   a) Definition of the problem parameters: geometric limits of the domain (L), Prandtl number (Pr), Rayleigh number (Ra) and temperature of the walls (T_{hot} and T_{cold}).
   b) Thermophysical properties of the fluid: heat capacity (c_p), dynamic viscosity (μ), density (ρ), thermal conductivity (λ) and coefficient of volumetric expansion (β).
   c) Initial pressure, velocity and temperature fields for the first iteration: P_{ini}, u_{ini}, v_{ini} and T_{ini}. All these values are set to 0 but the temperature, which is equal to $T_{ini} = 0.5$.
   d) Numerical parameters for the meshes discretization: N for the x-direction and M for the y-direction. The main mesh is $N \times M$.
   e) Arbitrary variables for choosing the convective interpolation schemes for the resolution of the momentum equation and the convection-diffusion equation.
   f) Numerical parameters for the solver: the same as for the Lid Driven Cavity algorithm, considering an additional relaxation factor for the temperature field Gauss-Seidel.
2. **Previous calculus.**
   a) Generation of the main, staggered-x and staggered-y meshes: definition of the geometry and assignation of the pressure, temperature and velocity fields. Pressure and velocity fields are analogous to the Lid Driven Cavity algorithm. The temperature field is defined at t^{n-1}, t^n and t^{n+1}, with an auxiliary field for the calculation of T^* in the Gauss-Seidel solver.
   b) Initialization of the discretization coefficients for the Poisson and convection-diffusion equations, where all the coefficients start at 0.

c) Initialization of time variable at $t = 0$ *sec*.

3. **Main loop.**
   a) Predictor velocity calculation: $\boldsymbol{R}(\mathbf{u})$ is calculated within a function, which takes into account the chosen interpolation scheme (CDS or QUICK). Equation (104) is considered for the staggered-y mesh, where the temperatures at the instants t^{n-1} and t^n are needed.
   b) Calculation of the coefficients for the Gauss-Seidel algorithm that solves the Poisson equation, which are defined in equation (93). The pressure boundary conditions are taken into account on the walls.
   c) Two-dimensional Gauss-Seidel algorithm for the pressure field.
      i. Apply equation (92) from the left to right, from the bottom to the top side nodes to calculate p^{n+1}, using p^* when the value of the scalar field is unknown and the last calculated value of p^{n+1} when possible. The relaxation factor (fr_1) is taken into account.
      ii. Is $max|p^{n+1} - p^*| < \delta$?
         A. If not, $p^* = p^{n+1}$ and go to (3ci).
         B. If yes, go to (3d).
   d) Calculation of the new velocity field by means of computing equations (94) and (95). The velocity boundary conditions on the walls are imposed on the respective nodes.
   e) Calculation of the coefficients for the Gauss-Seidel algorithm that solves the convection-diffusion equation, which are defined in equation (52). The temperature boundary conditions are taken into account on the walls.
   f) Two-dimensional Gauss-Seidel algorithm for the temperature field.
      i. Apply equation (51) from the left to right, from the bottom to the top side nodes to calculate T^{n+1}, using T^* when the value of the scalar field is unknown and the last calculated value of T^{n+1} when possible. The relaxation factor (fr_2) is taken into account.
      ii. Is $max|T^{n+1} - T^*| < \delta$?
         A. If not, $T^* = T^{n+1}$ and go to (3fi).
         B. If yes, go to (3g).
   g) Computation of the new time step by considering the CFL condition. The calculated time step is divided by the input parameter Q, as the user is able to modify it by hand in order to ensure the stability of the solver.
   h) Stop calculations and convergence criterion. Is $t < t_{max}$?
      i. If not, the solution is not converged within the specified maximum time. The loop is stopped and an error message is displayed on the screen.
      ii. If yes, is $max|u^{n+1} - u^n|$ AND $max|v^{n+1} - v^n| < \ tol$?
         A. If not, $t = t + \Delta t$, $u^{n-1} = u^n$, $u^n = u^{n+1}$, $v^{n-1} = v^n$ and $v^n = v^{n+1}$. Go to (3a).
         B. If yes, the solution is converged and the loop stopped. A message is displayed on the screed to notify it. The average Nusselt number is computed. Go to (4).

4. **Post-process.**
   a) Generation of output files with the geometry of the meshes, results of the simulation (pressure, temperature and velocity fields), the average Nusselt number and a list with the simulation data (input parameters and time until the converged solution).
   b) The evolution of the velocity fields at $L/2$ is plotted for different Rayleigh numbers. Plotting of the velocity streamlines, isolines and temperature isotherms. This is done by means of a MATLAB script.

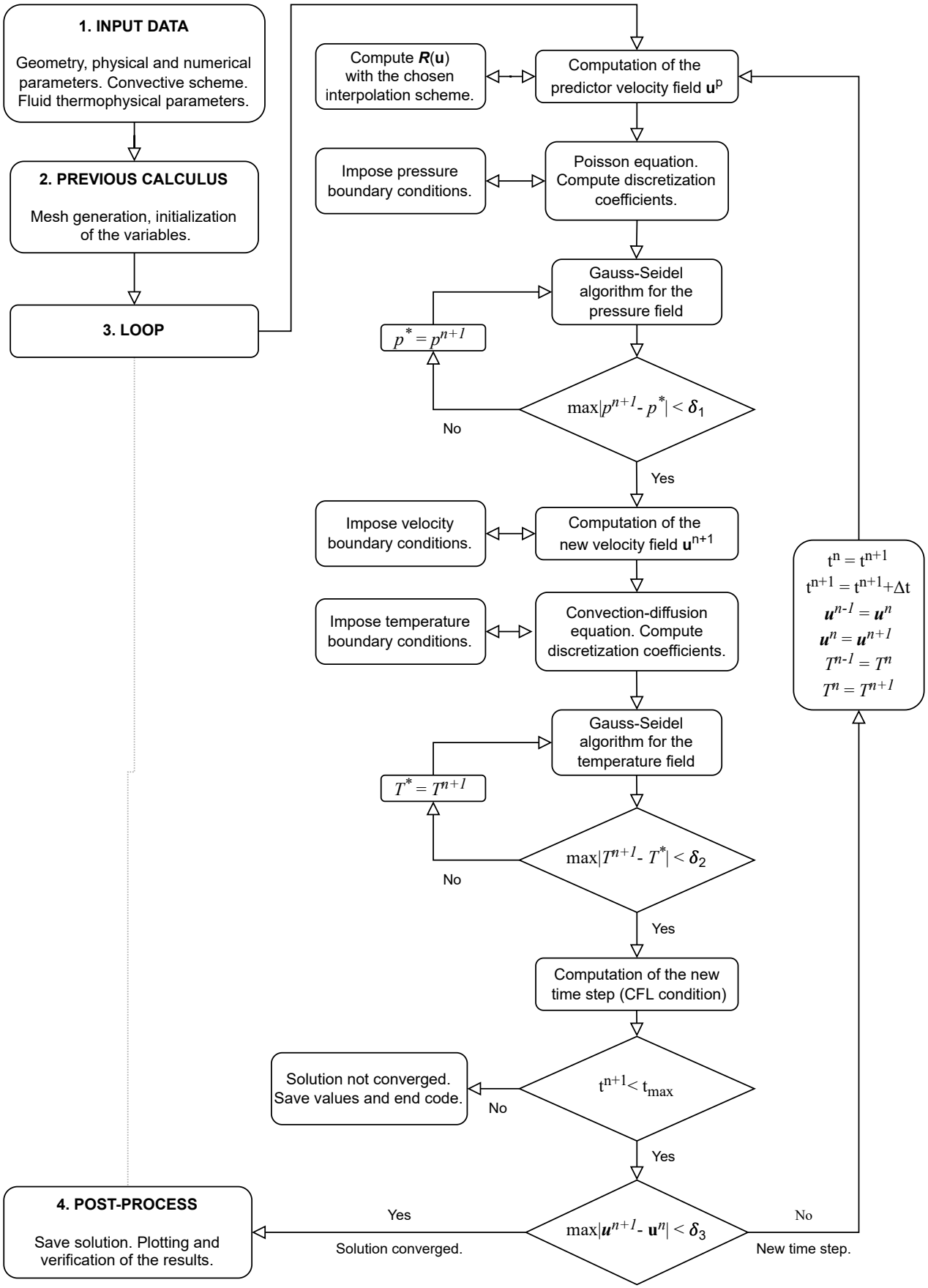


Figure 25: Differentially Heated Cavity. Resolution algorithm.

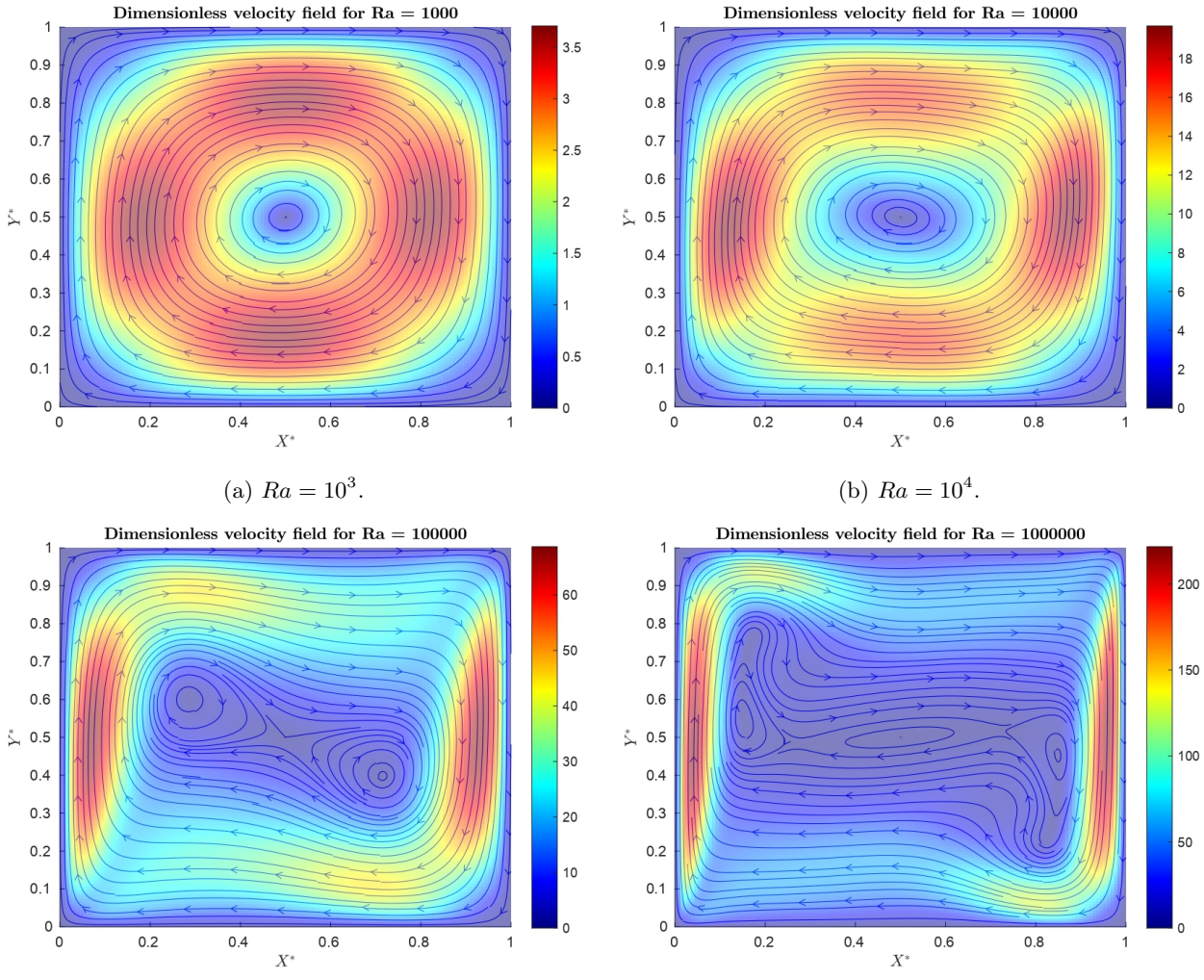
6.4. Results

The simulations are carried out for four different Rayleigh numbers: 10^3, 10^4, 10^5 and 10^6 with a 128×128 element mesh. The Prandtl number and the heat capacity are set to 0.71. The rest of thermophysical variables are set to 1, but the gravity, which is computed by taking into consideration the chosen Rayleigh number. Table 10 is a summary of the results obtained from the simulations, which is compared with the reference data [15] summarized in Table A.4 of the Appendix.

Ra	$\overline{Nu}$	$\frac{y}{L}\Big\vert_{x=L/2}$	u^*_{max}	$\frac{x}{L}\Big\vert_{y=L/2}$	v^*_{max}
10^3	1.114	0.816	3.650	0.176	3.699
10^4	2.237	0.824	16.184	0.121	19.627
10^5	4.513	0.855	34.780	0.066	68.680
10^6	8.848	0.848	65.192	0.0352	220.710

Table 10: Defferentially heated cavity. Simulation results.

The intensity of the velocity field and its streamlines are illustrated in Figure 26.



(a) $Ra = 10^3$.

(b) $Ra = 10^4$.

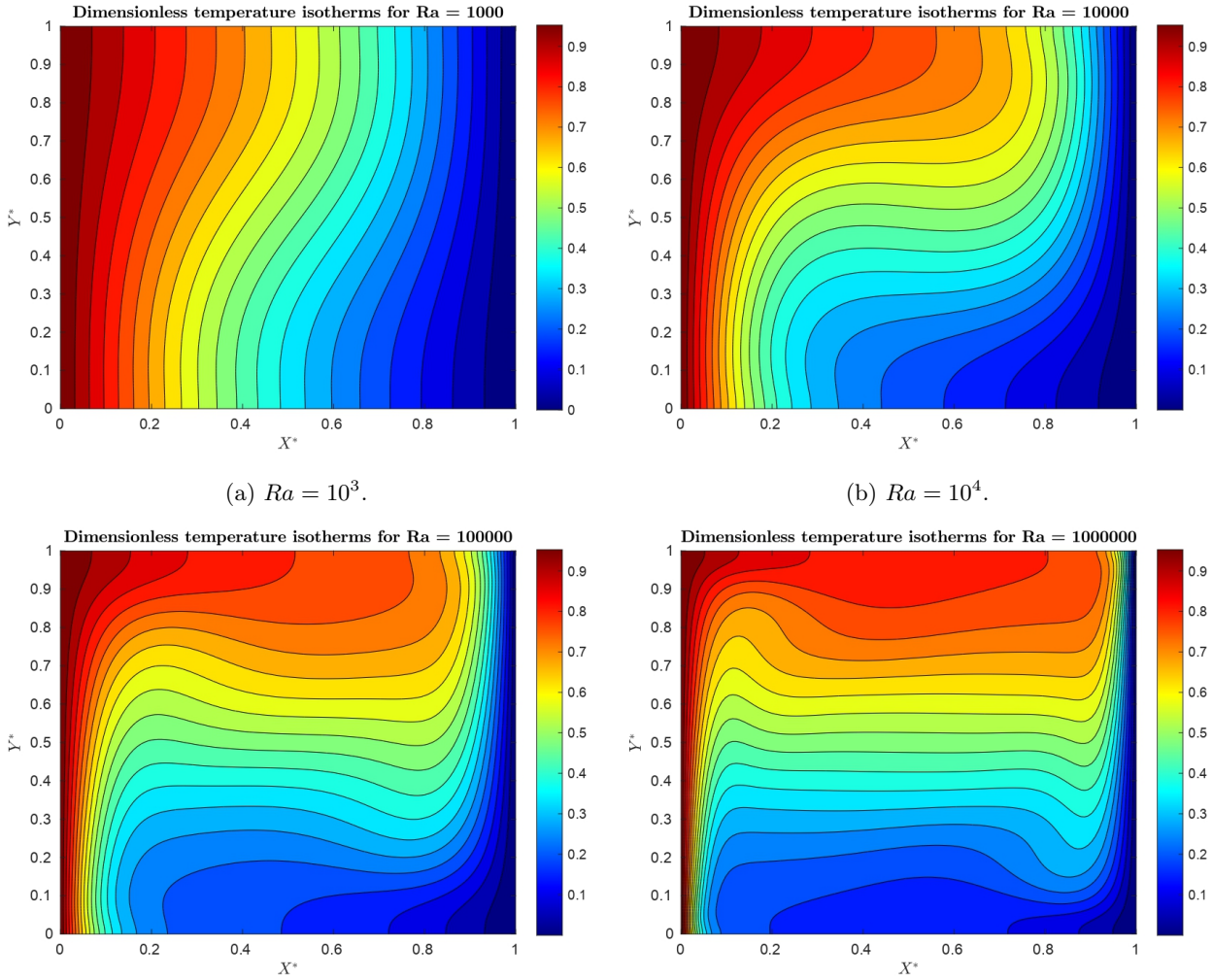
(c) $Ra = 10^5$.

(d) $Ra = 10^6$.

Figure 26: Differentially Heated Cavity. Velocity field intensity and streamlines for a 128x128 mesh.

The flow is moving in clockwise direction due to the combined effect of the temperature gradient and the vertical acceleration (gravity). For low Rayleigh numbers (10^3 and 10^4), a single vortex appears in the centre of the domain, and the horizontal and vertical velocities are relatively similar. When the Rayleigh number increases (10^5 and 10^6), the vertical velocity increases significantly more than the horizontal velocity, due to the increase of the vertical acceleration. This produces two additional vortexes on the northwest and southeast sides of the central vortex.

Regarding the temperature distribution, it is described in Figure 27.



(a) $Ra = 10^3$.

(b) $Ra = 10^4$.

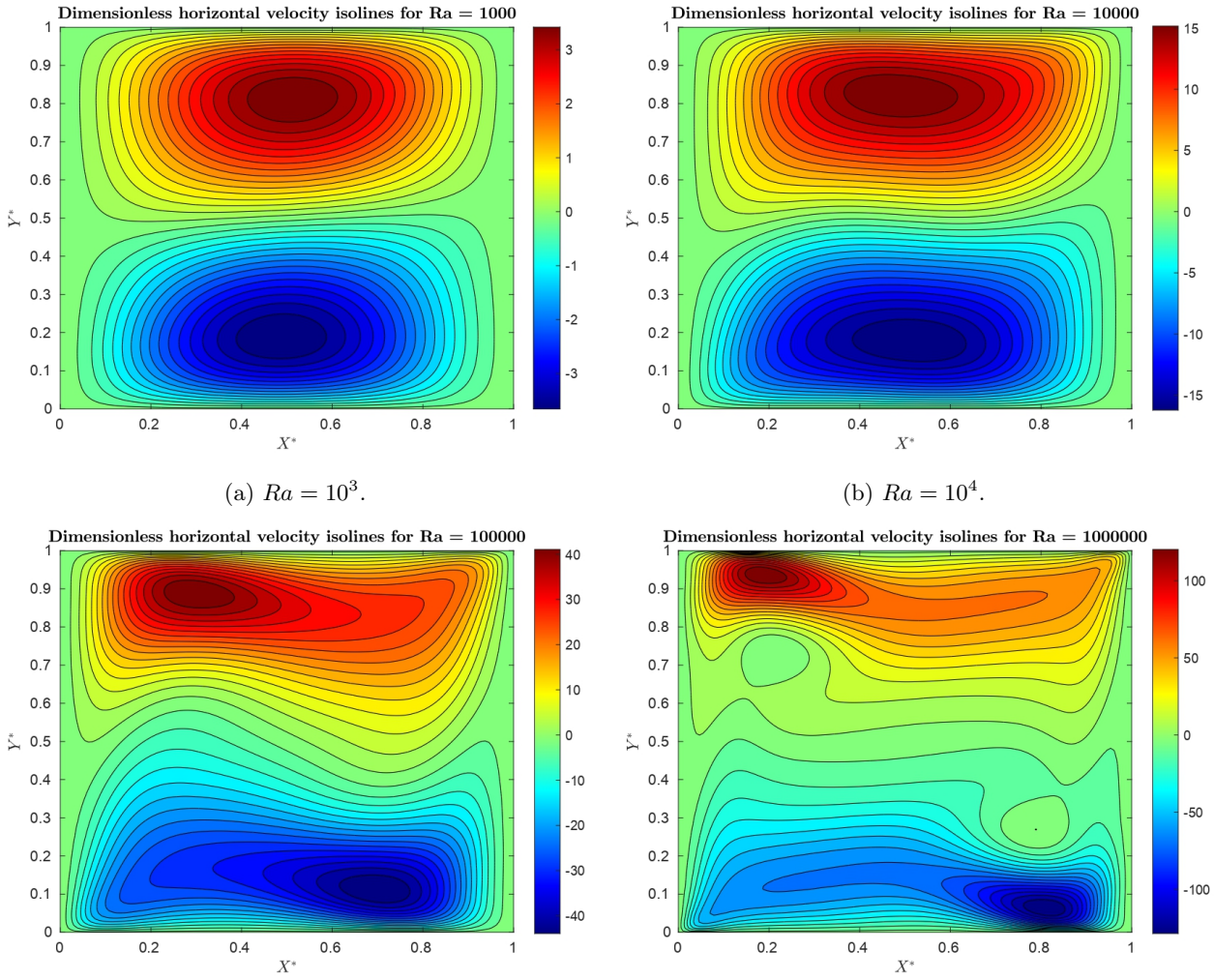
(c) $Ra = 10^5$.

(d) $Ra = 10^6$.

Figure 27: Differentially Heated Cavity. Temperature isotherms for a 128x128 mesh.

Hypothetically, if there was no gravity and the Rayleigh number was 0, the temperature gradient would be entirely in the x direction and the isotherms would look as vertical lines. As the Rayleigh number and the vertical acceleration increase, the flow gets to move faster in the bottom and top walls, increasing the convective effects nearby these locations. This produces a vertical temperature gradient, and the isotherm shapes that can be seen in Figure 27. For high Rayleigh numbers, the isotherms tend to be straight horizontal lines in the central positions of the geometry.

The horizontal and vertical velocity isolines are illustrated in Figures 28 and 29, respectively.



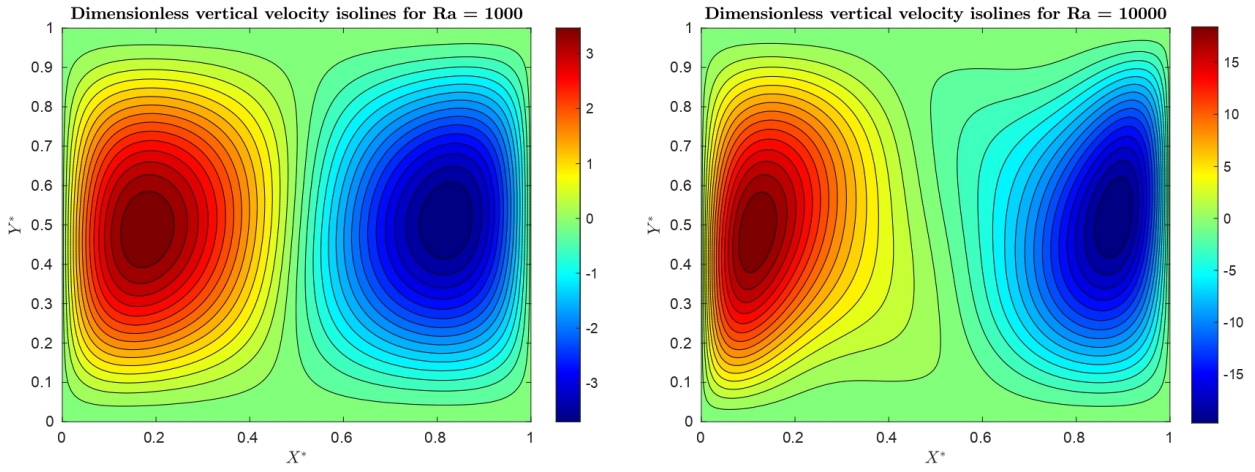
(a) $Ra = 10^3$.

(b) $Ra = 10^4$.

(c) $Ra = 10^5$.

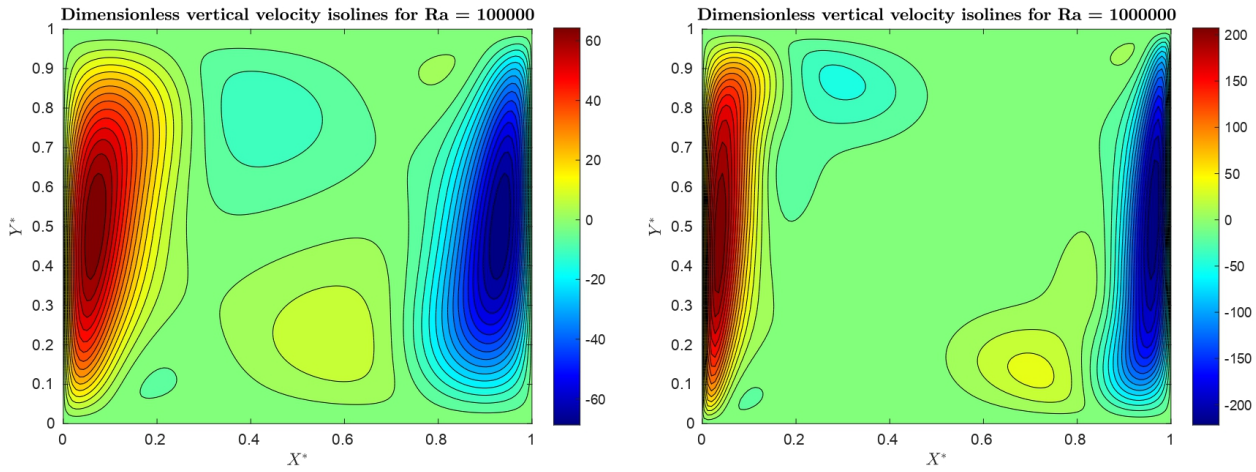
(d) $Ra = 10^6$.

Figure 28: Differentially Heated Cavity. Horizontal velocity isolines for a 128x128 mesh.



(a) $Ra = 10^3$.

(b) $Ra = 10^4$.



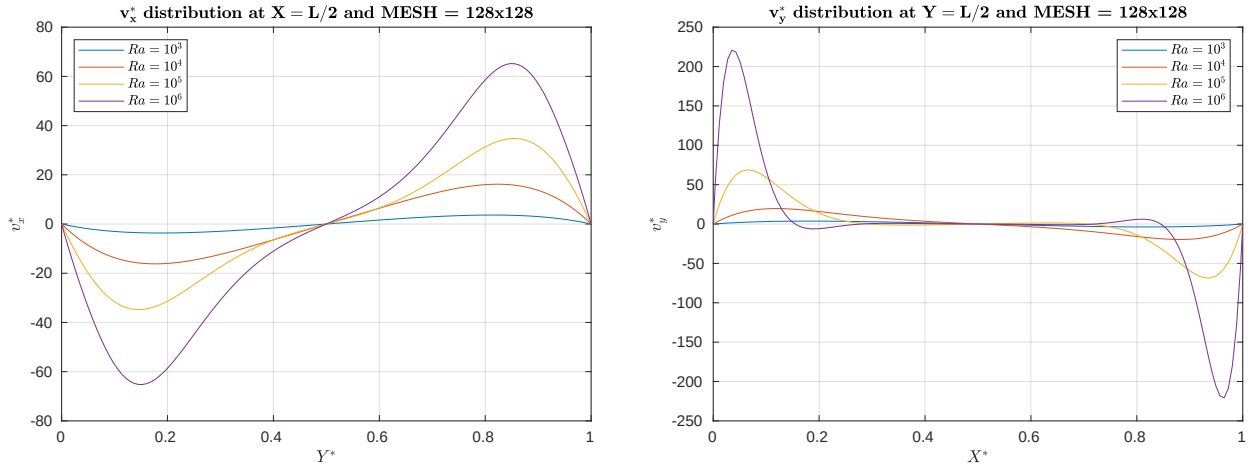
(c) $Ra = 10^5$.

(d) $Ra = 10^6$.

Figure 29: Differentially Heated Cavity. Vertical velocity isolines for a 128x128 mesh.

As expected due to the clockwise motion of the flow, the horizontal velocity is positive near to the top wall and negative on the bottom wall. On the other hand, the vertical velocity is positive in the hot wall and negative on the cold one. This effect becomes more important when the Rayleigh number increases.

Finally, Figure 30 shows the horizontal and vertical velocity distributions at $x^* = 0.5$ and $y^* = 0.5$, respectively. As mentioned above, the increase on the Rayleigh number increases the vertical acceleration and makes the flow go faster in both x and y directions.



(a) Horizontal velocity.

(b) Vertical velocity.

Figure 30: Differentially Heated Cavity. Velocity distributions for a 128x128 mesh.

7. The Square Cylinder Problem

7.1. Problem definition

The problem is defined as an uniform flow entering an open domain of dimensions $L \times H$, defined in Figure 31. A square cylinder of a very large depth and dimensions $D \times D$ is located at $x = l$ and $y = H/2$. The objective is to find the velocity and pressure distributions of the fluid moving across this domain.

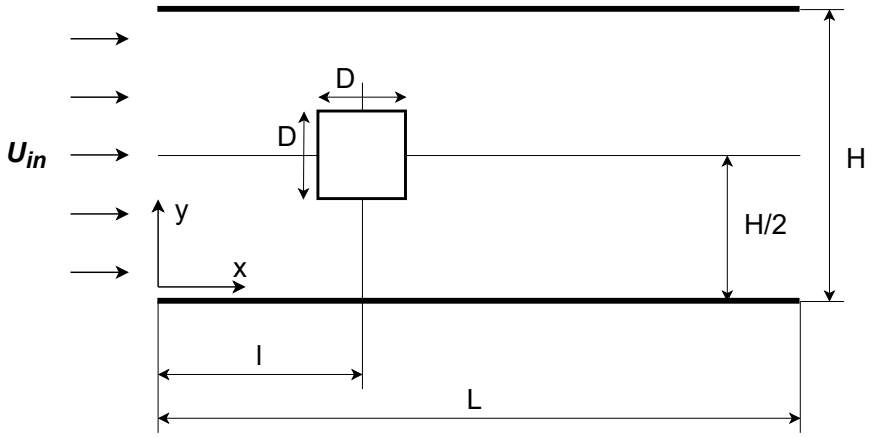


Figure 31: Illustration of the square cylinder problem geometry.

For this specific dissertation, the values of the dimensions and parameters are defined as follows:

L [m]	H [m]	l [m]	D [m]	U_{in} [m/s]
30	8	10	1	1

Table 11: Square Cylinder. Problem parameters and dimensions of the geometry.

7.2. Practical considerations

7.2.1. Mesh definition and boundary conditions

The method employed in solving the problem is the Fractional Step Method. Due to this, the algorithm for computing the velocity and pressure fields is completely analogous to the algorithm of the Lid Driven Cavity problem. The only thing to take into consideration is the treatment of the square cylinder and the corresponding boundary conditions.

First of all, a non-equidistant mesh has been implemented in order to improve the accuracy of the results. The strategy to do so is to divide the domain in nine different zones with a different and equidistant mesh each, as it is described in Figure 32. This permits to insert nodes on the faces of the cylinder in the staggered meshes, and therefore apply the velocity no-slip boundary condition ($u = 0\ m/s,\ v = 0\ m/s$ on the cylinder walls). Regarding the inner nodes of the cylinder, the velocity is set to 0 to all the nodes.

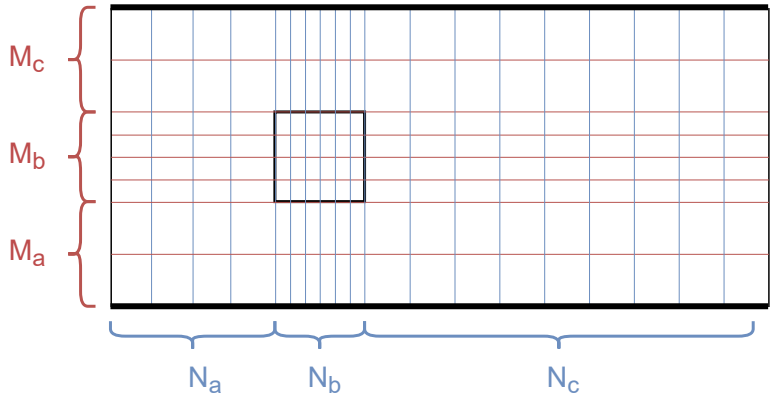


Figure 32: Square Cylinder. Illustration of the main mesh discretization.

Regarding the considered boundary conditions on the external walls of the domain, they are stated below. Just like in the Lid Driven Cavity problem, the velocity boundary conditions are imposed on the computation of $\mathbf{u}^{n+1}$, and the pressure ones on the calculation of the discretization coefficients of the Poisson equation.

$$\text{Left wall:} \quad u = U_{in}, \quad v = 0, \quad \frac{\partial p}{\partial x} = 0. \tag{110}$$

$$\text{Right wall:} \quad \frac{\partial u}{\partial x} = 0, \quad \frac{\partial v}{\partial x} = 0, \quad p = 0. \tag{111}$$

$$\text{Top wall:} \quad \frac{\partial u}{\partial y} = 0, \quad v = 0, \quad \frac{\partial p}{\partial y} = 0. \tag{112}$$

$$\text{Bottom wall:} \quad \frac{\partial u}{\partial y} = 0, \quad v = 0, \quad \frac{\partial p}{\partial y} = 0. \tag{113}$$

The pressure boundary conditions of the main mesh nodes in contact with the cylinder are a particular case, which also need to be imposed on the discretization coefficients of the Poisson equation. As there is no mass flow rate in the normal direction of the walls of the cylinder, all the coefficients are as described in equation (93) but the one corresponding to the wall, which is set to 0. For instance, in Figure 33 the east coefficient is imposed to be zero ($a_E = 0$).

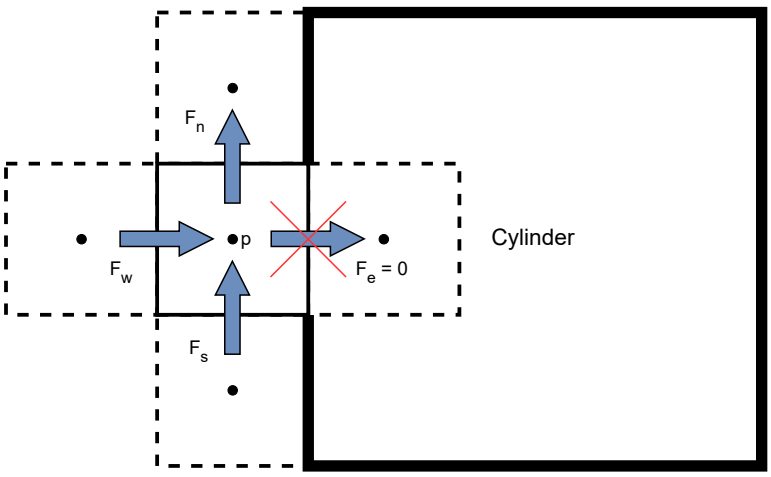


Figure 33: Square Cylinder. Main mesh control volume in contact with the cylinder.

In regard to the pressure boundary conditions of the inner nodes of the cylinder, all the coefficients are set to 0 but a_P, which is set to 1. This imposes the pressure field to be zero inside the cylinder at all times, despite the final solution does not depend on this fact.

7.2.2. Dimensionless coefficients

The lift coefficient (C_l), drag coefficient (C_d) and Strouhal number (St, for the non-steady cases) are three important dimensionless coefficients that define the behaviour of the problem [16]. These are given by:

$$C_l = \frac{2F_l}{\rho U_{in}^2 D}, \tag{114}$$

$$C_d = \frac{2F_d}{\rho U_{in}^2 D}, \tag{115}$$

$$St = \frac{f_v D}{U_{in}}, \tag{116}$$

where F_l and F_d are the lifting and resistance forces seen by the square cylinder, respectively. The parameter ρ corresponds to a reference density defined by the user in the input of the code, and f_v is the vortex-shedding frequency. This last parameter is calculated by means of a Power Spectral Density (PSD), employing the Fast Fourier Transform (FFT) algorithm of MATLAB over the time series of the lift coefficient, $C_l(t)$.

Regarding the lifting and resistance forces, they are caused by pressure and tangential forces over each differential element of area of the body, a diagram is included in Figure 34 to better understand the phenomenology. Their equations are of popular knowledge, and deduced from [17].

$$F_l = \int_A dF_y = -\int_A p \sin\theta dA + \int_A \tau_w \cos\theta dA \tag{117}$$

$$F_d = \int_A dF_x = +\int_A p \cos\theta dA + \int_A \tau_w \sin\theta dA \tag{118}$$

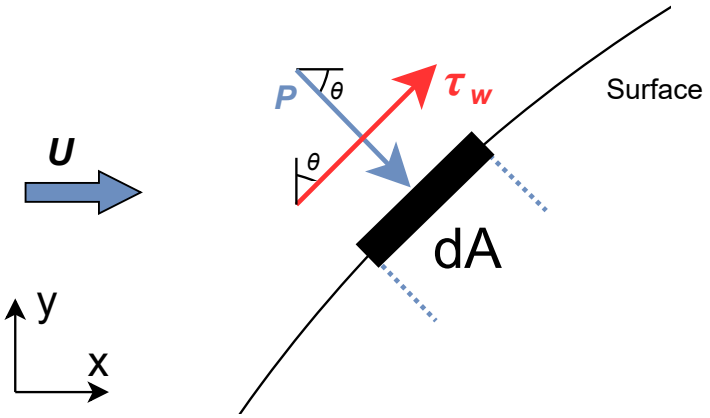


Figure 34: Square Cylinder. Aerodynamic forces over an arbitrary surface.

The stress tensor ($\boldsymbol{\tau}$) for Newtonian fluids is described by the Stokes's viscous law, which in its vectorial formulation is stated as follows:

$$\boldsymbol{\tau} = \mu \left(\nabla \mathbf{u} + \nabla \mathbf{u}^T \right) - \frac{2}{3} \mu \left(\nabla \cdot \mathbf{u} \right) \boldsymbol{\delta} \tag{119}$$

where $\boldsymbol{\delta}$ is an identity matrix. In two dimensions, the tangential stress[10] is described by the following equation (120), which is discretized into equation (121). Note that the viscosity must be calculated by means of the reference density and the Reynolds number, $\mu = \frac{\rho U_{in} D}{Re}$.

$$\tau_w = \mu \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) \tag{120}$$

$$\tau_w = \mu \left(\frac{u_n - u_s}{\Delta y} + \frac{v_e - v_w}{\Delta x} \right) \tag{121}$$

Finally, as mentioned before the algorithm for the resolution of this problem is completely analogous to the algorithm of the Lid Driven Cavity. For this reason, there is no further explanation nor diagram regarding the algorithm of the present problem. Despite this, the main differences with the Lid Driven Cavity case are the above explained modification of the mesh, the treatment of the boundary conditions, and a function that calculates the lift and drag coefficients each converged iteration.

7.3. Results

The simulation is carried out for different Reynolds numbers from $Re = 1$ to $Re = 200$. It has been observed that, for Reynolds numbers lower than 50, steady solutions are found. For higher values of Reynolds number, a periodic vortex-shedding effect appears within the wake, which produces a periodic lifting force centered in $C_l = 0$. For this reason, the Strouhal number represents well the solution only for the non-steady cases, while the steady cases are better represented by the drag coefficient (as there exists no lifting force).

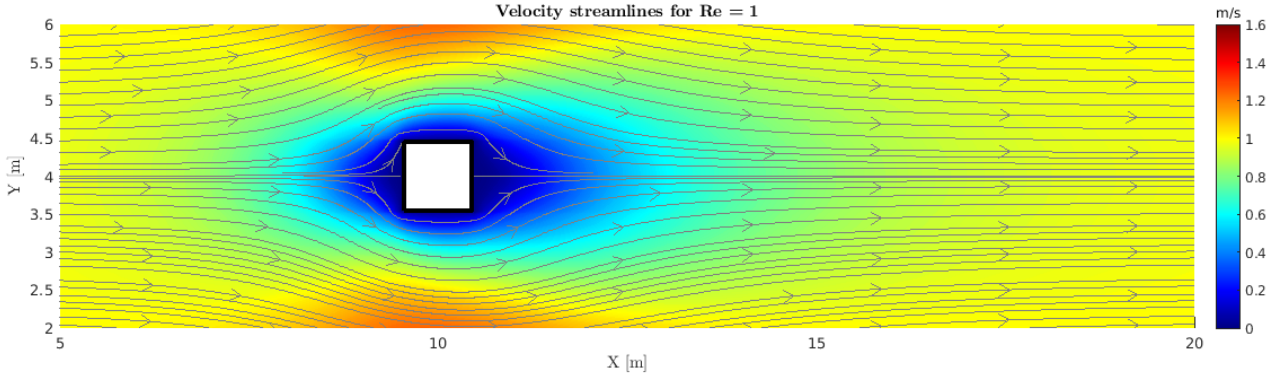
All the simulations have been carried out with the same mesh size, with horizontal divisions of $N_a = 40$, $N_b = 20$, $N_c = 80$ and vertical divisions of $M_a = 25$, $M_b = 35$, $M_c = 25$. This yields a mesh of 140×85 elements, corresponding to a total of $11,900$ elements.

First of all, the streamlines and velocity intensity plots around the cylinder are included in Figure 35. It is important to note that only the cases with $Re = 1$, $Re = 25$, $Re = 100$ and $Re = 200$ are taken into account for this figure, being the first two cases steady. The representation of the non-steady cases are instantaneous plots at $t = 300$ *sec*, as the vortex-shedding can be appreciated.

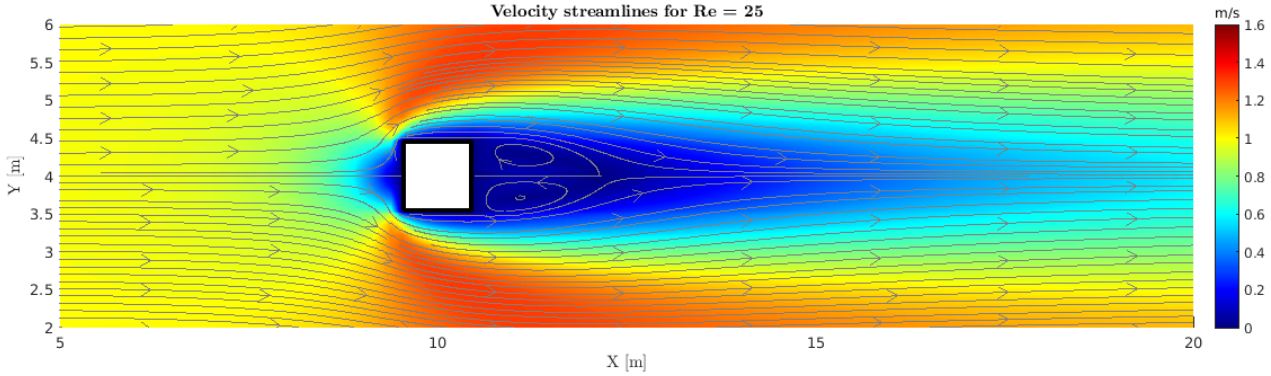
For $Re = 100$, a video of the evolution of the velocity field module has been recorded. It can be found at the following hyperlink: VIDEO. The transient evolution and appearance of the vortex-shedding can be appreciated. Note that $t = 0$ *sec* is set relatively to the velocity distribution of the first photogram of the video.

The module of the velocity and pressure distributions are included in Figures 36 and 37. Note that, while the streamlines plots of Figure 35 are zoomed nearby the area of the cylinder, the rest of plots are represented all over the domain.

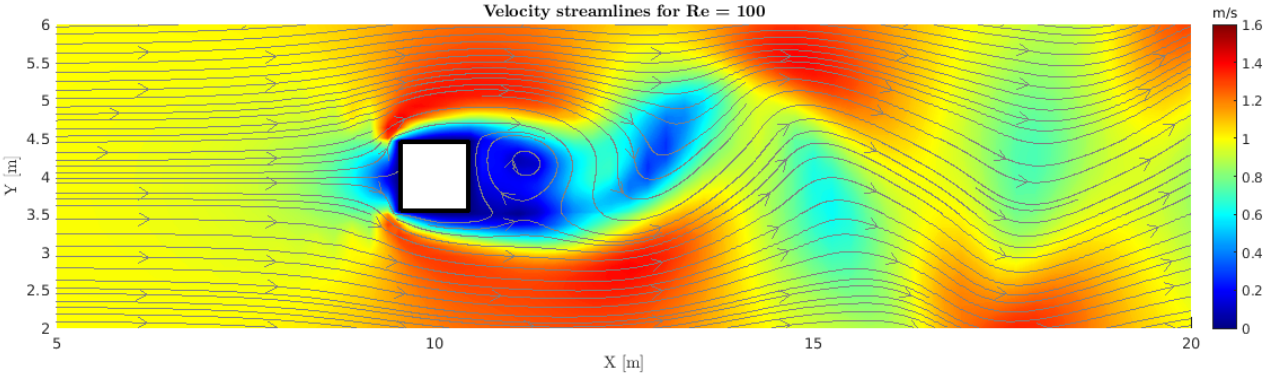
[10] τ_w corresponds to the component τ_{xy} of the stress tensor.



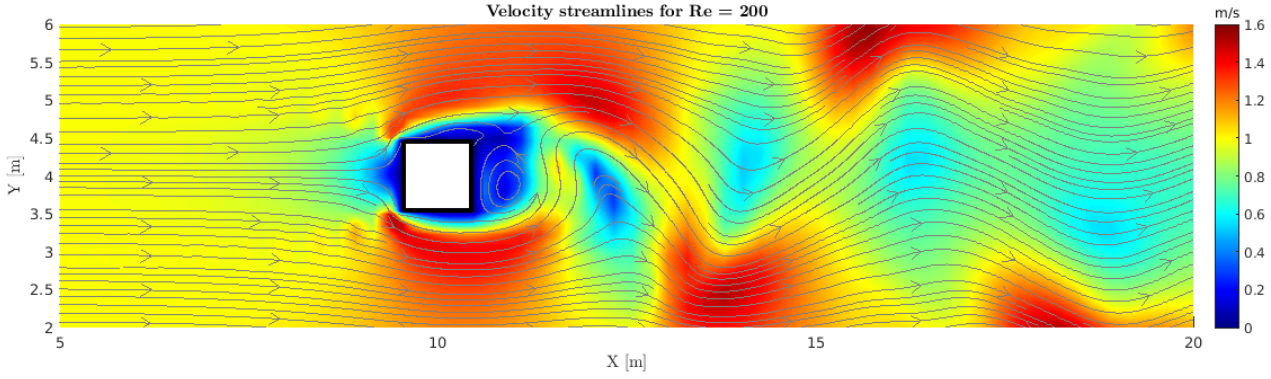
(a) Steady solution for $Re = 1$.



(b) Steady solution for $Re = 25$.

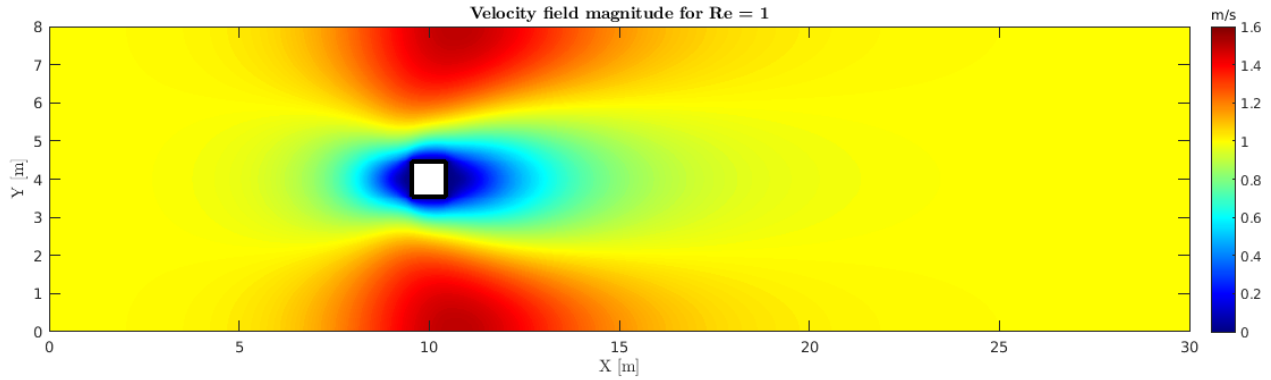


(c) Non-steady solution for $Re = 100$.

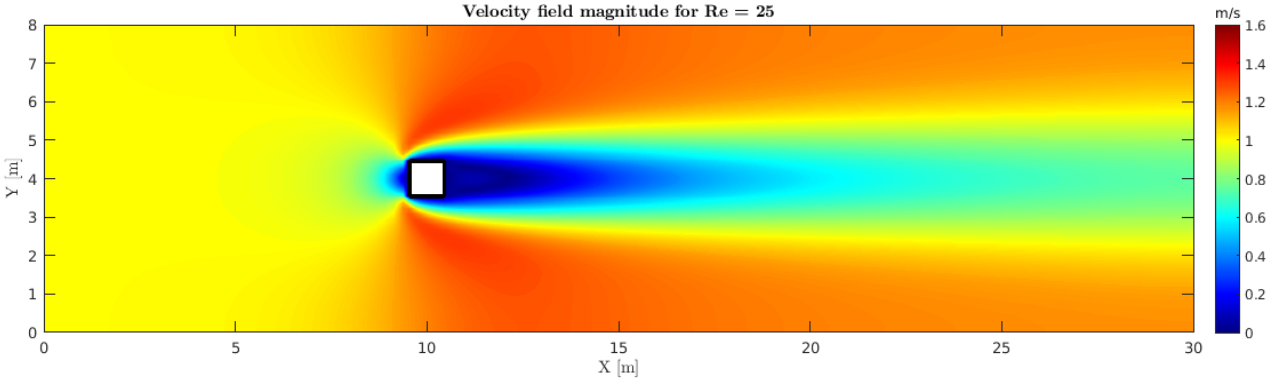


(d) Non-steady solution for $Re = 200$.

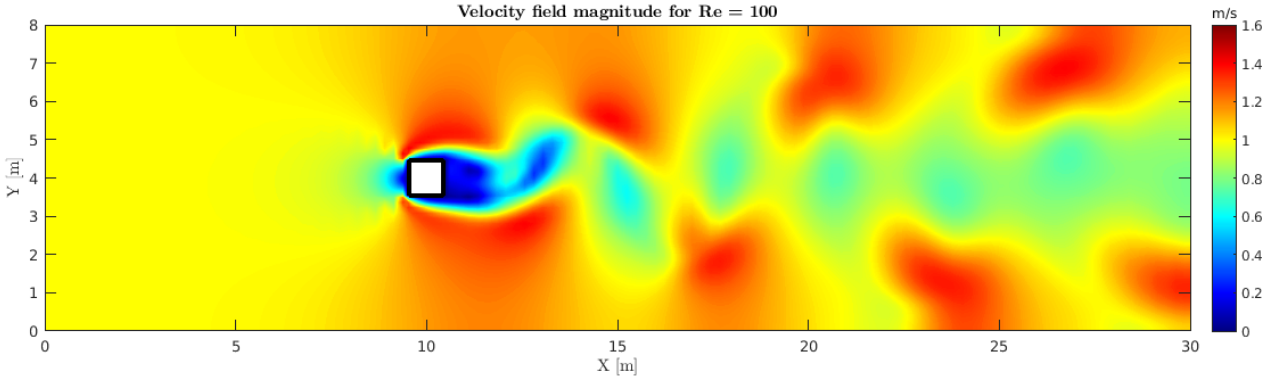
Figure 35: Square Cylinder. Zoom of the velocity streamlines and module for different Reynolds numbers.



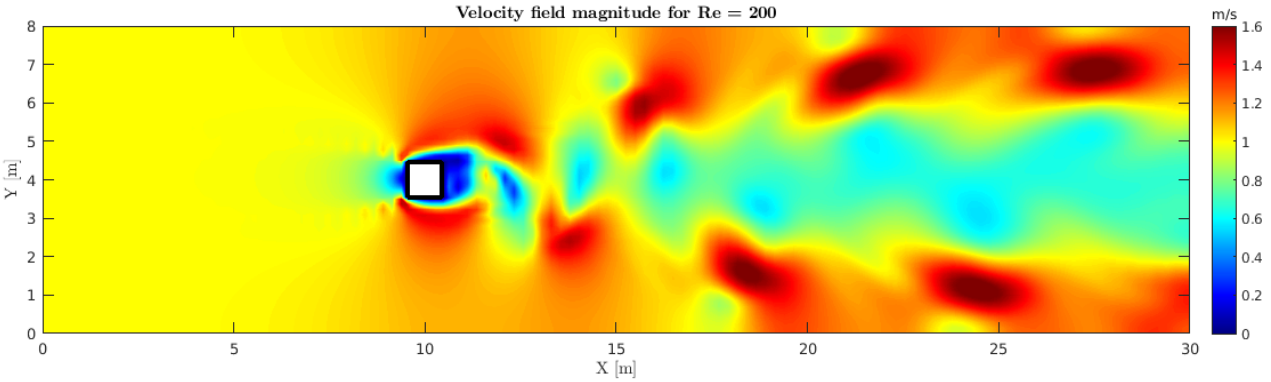
(a) Steady solution for $Re = 1$.



(b) Steady solution for $Re = 25$.

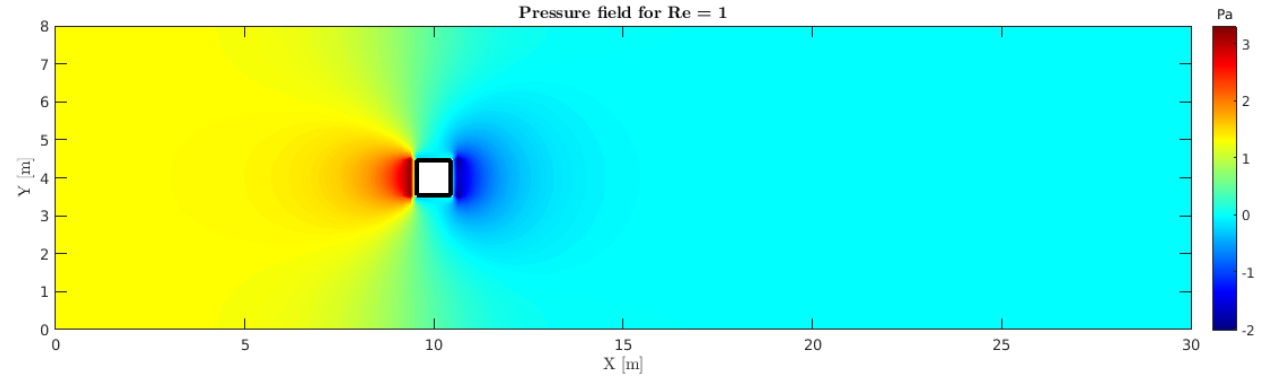


(c) Non-steady solution for $Re = 100$. See VIDEO.

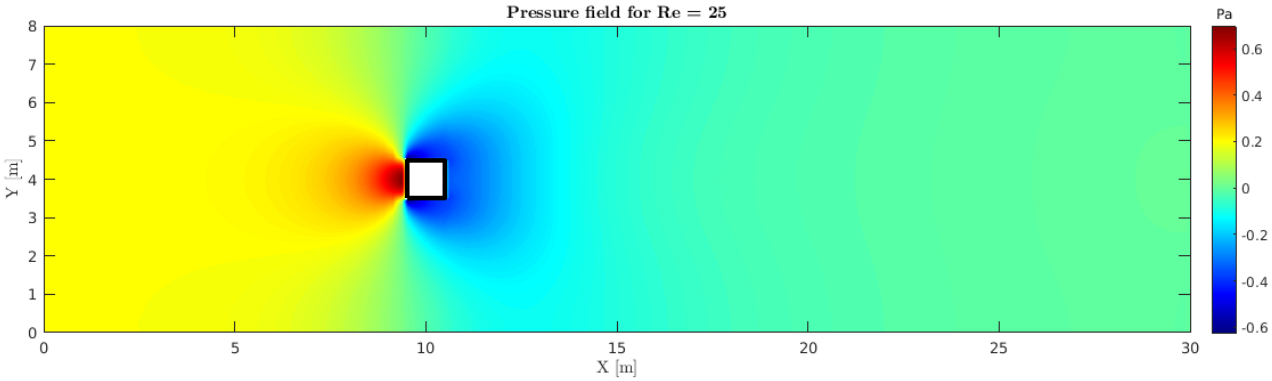


(d) Non-steady solution for $Re = 200$.

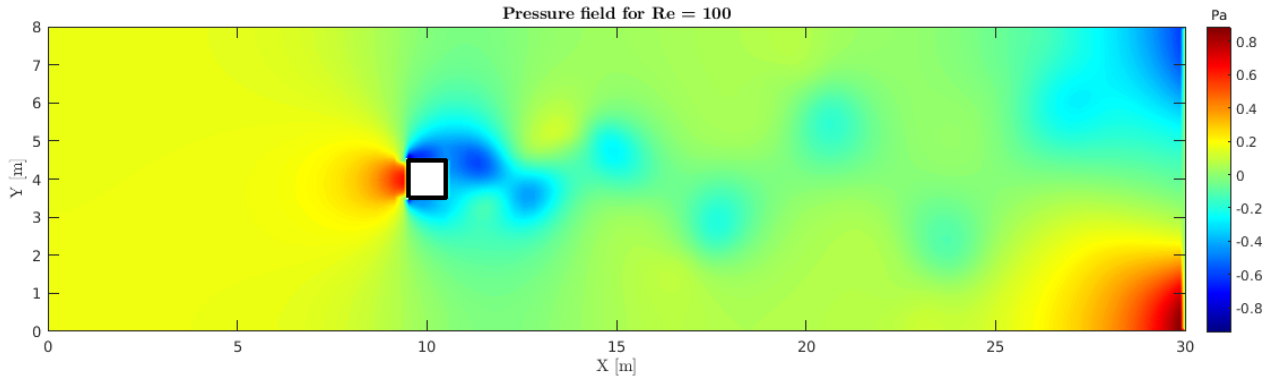
Figure 36: Square Cylinder. Velocity module for different Reynolds numbers.



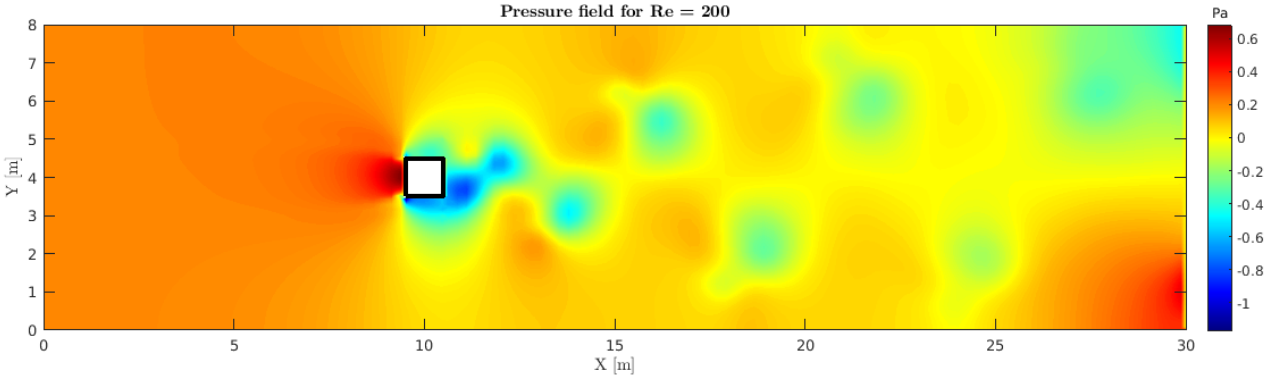
(a) Steady solution for $Re = 1$.



(b) Steady solution for $Re = 25$.



(c) Non-steady solution for $Re = 100$.



(d) Non-steady solution for $Re = 200$.

Figure 37: Square Cylinder. Pressure field intensity for different Reynolds numbers.

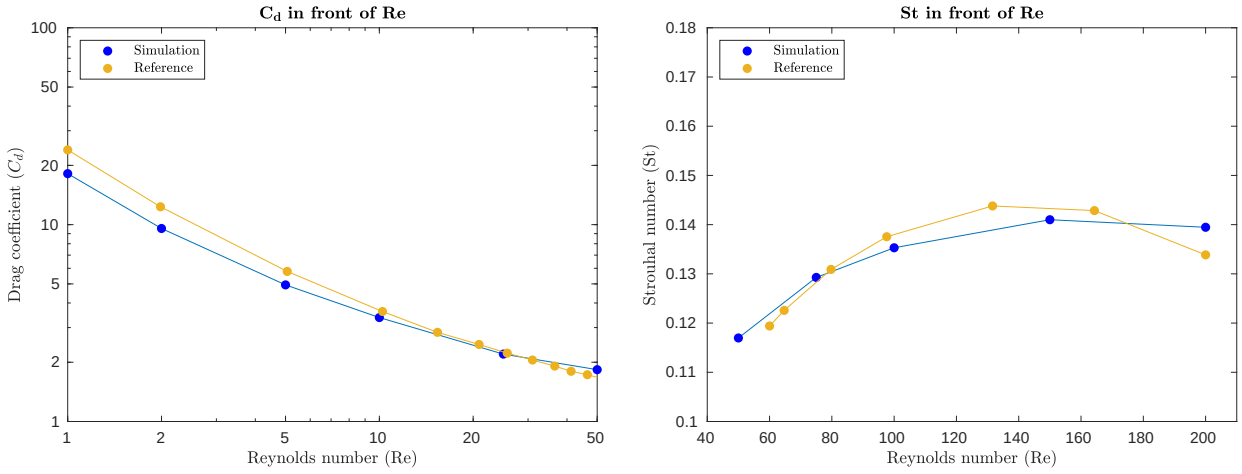
Regarding the velocity field distribution, for $Re = 1$ a completely symmetric solution is found. The diffusion effects are strong, smoothing the solution and avoiding sudden gaps in velocity field intensity. The flow persists without separation at the trailing edges of the sharp-edged body. For $Re = 25$, the solution is also steady but two vortexes appear in the wake due to the separation of the laminar boundary layers over the trailing edges of the cylinder. These vortexes are symmetric and located in the recirculation region of the wake just after the cylinder. As it can be seen, the streamlines gather on the spots where the velocity intensity is larger.

When increasing the Reynolds number, the vortex-shedding effects appear and the solution becomes unsteady, this phenomenon is well known as the von Kármán vortex street. The critical Reynolds number when the near-wake becomes unstable is set at $Re_{crit} = 54$ by Klekar and Patankar [18]. The behaviour of the solution is analogous for $Re = 100$ and $Re = 200$: a large dynamic vortex appears just after the square cylinder, which generates oscillations on the wake.

These oscillations get larger with the Reynolds number and the decrease of viscous forces, increasing the lifting force seen by the cylinder and the frequency of the shedding-vortexes. The separation point of the boundary layer moves from the trailing edge to the leading edge around $Re \approx 150$ [19], which causes a decrease on the frequency of the vortex and the Strouhal number with the Reynolds number. This behaviour can be observed in Figure 38b, as the maximum point of the Strouhal number is located around $Re \approx 150$.

In regard to the pressure distribution, the stagnation point on the left side of the cylinder makes the flow to suddenly stop, increasing the pressure on this spot. For low Reynolds numbers, the pressure gradient within the left and right faces of the cylinder is strong, generating high values of drag coefficient. As the viscous forces decrease, this pressure gradient also diminishes, reducing the drag coefficient with the Reynolds number. For high values or Re, despite having an oscillating behaviour, both viscous and pressure contributions to the drag coefficient get to decrease. This effect can be seen in Figure 38a.

The results of the simulation are compared with digitized results extracted from reference [19]. On the one hand, the evolution of the drag coefficient with the Reynolds number represents well the results of the simulation for low Re, as its calculation involves both velocity and pressure fields, and the Strouhal number is 0 by definition. On the other hand, the Strouhal number evolution in front of Re describes well the behaviour of the solution for high Reynolds numbers, as the frequency of the vortex-shedding is strictly bonded to the pressure and velocity fields.



(a) Steady solutions. Low Reynolds numbers.

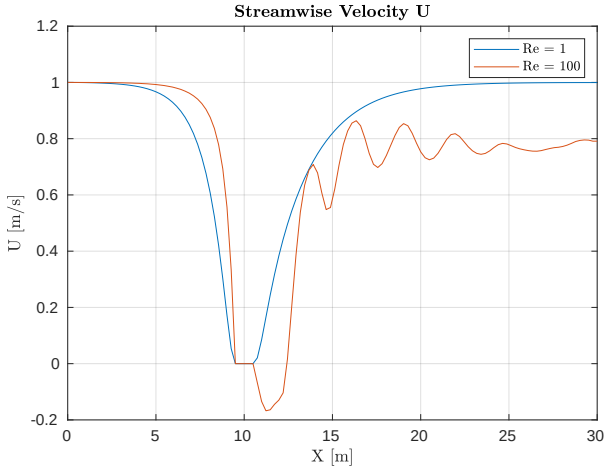
(b) Non-steady solutions. High Reynolds numbers.

Figure 38: Square Cylinder. Drag coefficient and Strouhal number in front of Re.

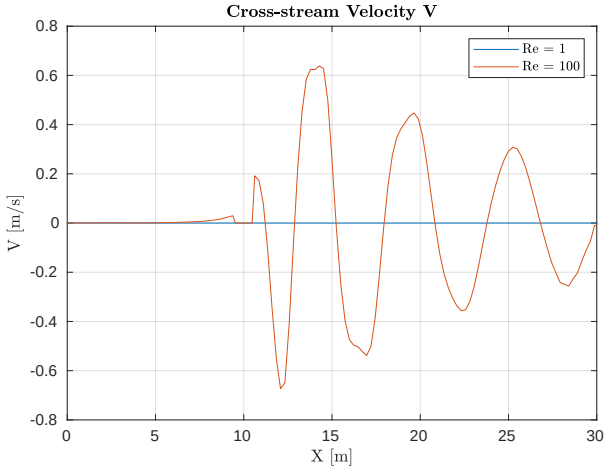
As it can be seen, the results of the simulation are highly correlated with the reference. Despite this, some slight differences are appreciated, probably caused by the mesh size or minor errors on the code.

Finally, the horizontal and vertical velocities are represented along the x and y directions, as represented in Figure 39. A VIDEO is included to see the evolution of these velocities in time. Regarding the x-direction, the evolution of the velocities is shown at $y = 4\ m$ for $Re = 1$ and $Re = 100$. It is interesting to observe the velocity oscillations caused by the vortexes ($Re = 100$), and compare them to a completely stable and steady case ($Re = 1$). Furthermore, the velocity turns to zero at $x \geq 9.5\ m$ and $x \leq 10.5\ m$, as the flow finds the cylinder on its path.

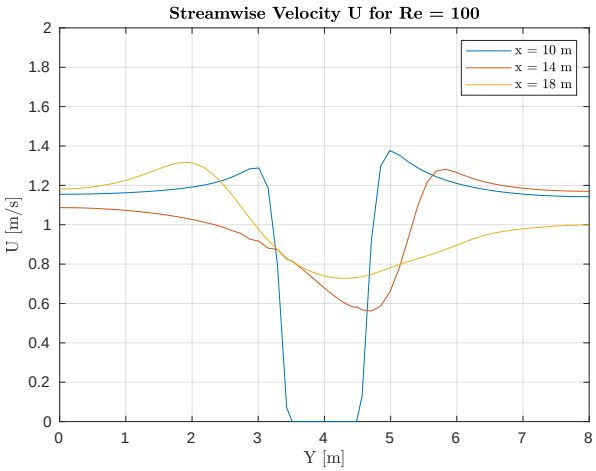
Regarding the y-direction, the results are represented at $x = 10\ m$, $x = 14\ m$ and $x = 18\ m$ for $Re = 100$. On the first, the velocities turn to zero as they find the cylinder at $y \geq 3.5\ m$ and $y \leq 4.5\ m$. For $x = 14\ m$ and $x = 18\ m$, the peaks of the curves depend on the position of the vortexes at the precise moment of the representation. Reference plots extracted from [19] are shown in Figure A.1 of the Appendix. Note that, despite having the same behaviour and magnitude, remarkable differences appear due to the different moment of representation and the parabolic flow of the reference. For this reason, the simulation and reference plots are not superposed in the report.



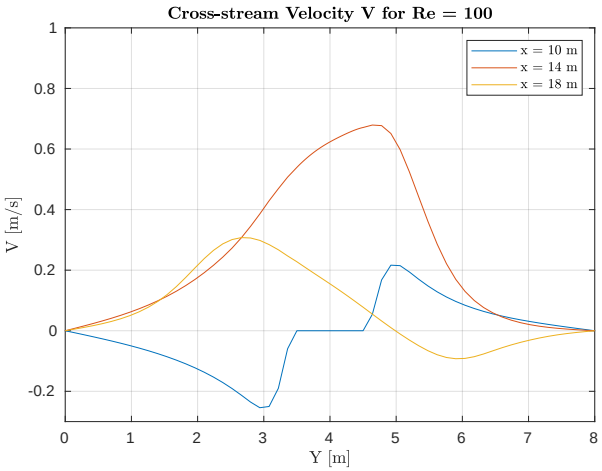
(a) Horizontal velocity in the x-direction, at $y = 4\ m$.



(b) Vertical velocity in the x-direction, at $y = 4\ m$.



(c) Horizontal velocity in the y-direction.



(d) Vertical velocity in the y-direction.

Figure 39: Square Cylinder. Simulation of the velocity distribution at different positions. See VIDEO.

8. Introduction to Turbulence

8.1. Chaos, determinism and nonlinearity

Turbulence is the word used to describe a fluid that appears to behave randomly. The governing equations of this fluid (the Navier-Stokes equations) are deterministic, chaotic and nonlinear. A chaotic system is a deterministic system that must be nonlinear [20].

Deterministic means that, given the necessary initial conditions, the movement of the fluid can be predicted in time and space by solving the Navier-Stokes equations. The fact of being chaotic means that the system is highly sensitive to the initial conditions, obtaining completely different solutions in time by changing these initial conditions differentially.

The origin of turbulence is found in the only nonlinearity source of the Navier-Stokes equations: the convective term of the momentum equation. This one induces complexity in terms of small scale phenomena, which are finally dissipated in the form of energy.

The largest eddies of a system conform the driving scale, which is under highly convective effects and an elevated Reynolds number. These generate smaller eddies iteratively, until they are dissipated when diffusion effects are on the same order of magnitude than the convective effects. These small eddies add complexity to the solution, increasing the computing effort necessary to solve the Navier-Stokes equations numerically.

8.2. Kolmogorov energy spectrum

A turbulent system is a combination of phenomena of large and small scales. These are defined by their energy distribution in front of the wave number k, inversely proportional to the wave length of the eddy. As k increases, the eddies are smaller and less energetic in terms of kinetic energy. This can be seen in Figure 40, which corresponds to the mentioned energy distribution in logarithmic scale.

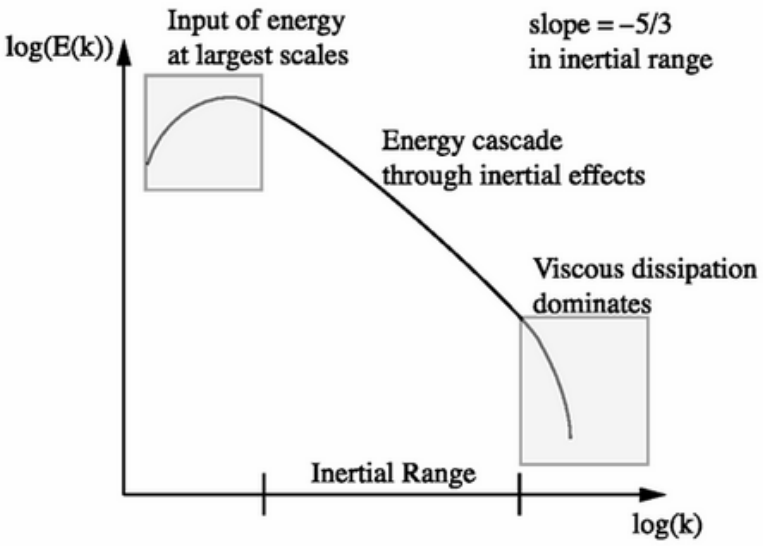


Figure 40: Illustration of the energy cascade [21].

Three different zones can be identified. First of all, the energetic zone on the lower values of k. This is characterized by having a time-dependant external input of energy (ε_{in}), which generate the largest scales of the problem, also called driving scales. The convective effects are dominant on this zone, which has the largest Reynolds number.

The second zone is the inertial range, composed of a straight line with a negative slope of $-5/3$. Kolmogorov found an expression for this line, defined in equation (122) known as Kolmogorov energy spectrum. This expression is based on his hypothesis number 41 (K41), which states that the distribution of energy is only proportional to the input of energy and the wave number ($E(k) \propto \varepsilon^a k^b$). The exponents a and b can be easily found by means of a dimensional analysis.

$$E(k) = C_k \varepsilon^{2/3} k^{-5/3} \tag{122}$$

The coefficient C_k is approximately $C_k \approx 1.4$, and the only parameter different between one turbulent flow to another is ε [21]. In the inertial range, the larger scales induce smaller scales in cascade, diminishing the Reynolds number and the convective effects.

The third zone is where the dissipation of the smallest scales occurs. It is called Kolmogorov length scale, and it consists on the less energized eddies that are dissipated due to the relative increase of the viscous forces, which are on the same order of magnitude than the inertial forces. Therefore, the Reynolds number is in the order of magnitude of $Re \sim 1$. This dissipation generates a time-dependant output of energy (ε_{out}), which by hypothesis of equilibrium must be equal of the input of energy ($\varepsilon_{in} = \varepsilon_{out}$).

8.3. Computational complexity of a DNS

In this section, the computational complexity of a Direct Numerical Simulation (DNS) in a turbulent system is studied. First of all, it is necessary to define a nomenclature. For the largest eddies, their associated velocity and wavelength are set to U and l, while u and μ stand for the smallest eddies, respectively.

The next step is to express the input/output of energy and the Reynolds number in terms of the latest parameters and the kinematic viscosity. In regard to the input and output of energy, they are in the order of $\varepsilon_{in} \sim U^2/(l/U) = U^3/l$ and $\varepsilon_{out} \sim \ \nu\left(u^2/\mu^2\right)$, which stand for energy over time. The Reynolds number of the largest eddies is $Re_l = Ul/\nu$, and for the smallest eddies $Re_\mu = u\mu/\nu \sim 1$. From the hypothesis of equilibrium, it is possible to obtain relation (123).

$$\varepsilon_{in} = \varepsilon_{out} \ \longrightarrow \ U^3/l \sim \nu\left(u^2/\mu^2\right) \tag{123}$$

Introducing the definition of Re_l and Re_μ into (123), the following set of expressions (124) is obtained. Note that t_l and t_μ are defined as l/U and μ/u, respectively. These are the characteristic times of the largest and smallest eddies.

$$\begin{aligned} \mu/l &\sim Re_l^{-3/4} \\ u/U &\sim Re_l^{-1/4} \\ t_\mu/t_l &\sim Re_l^{-1/2} \end{aligned} \tag{124}$$

At this point, the decrease on the wavelength, velocity and characteristic time for the smallest scales as a function of the Reynolds number of the driving scales is already known. When the smaller scales appear and μ/l decreases by an arbitrary factor, the mesh must be refined in space and time to be able of fitting these effects. This adds computational complexity to the numerical resolution of the problem, increasing the computing effort at least in the order of the inverse of expressions (124).

- In a 3-D space: $(l/\mu)^3 \sim Re_l^{9/4}$
- In time: $t_l/t_\mu \sim Re_l^{1/2}$
- In space $\times$ time: $(l/\mu)^3 \, t_l/t_\mu \sim Re_l^{11/4}$

To conclude, the computational complexity of the Navier-Stokes equations grows in the order of $Re_l^{11/4}$. Solving a turbulent flow of high Reynolds number with a DNS is very expensive in terms of computational effort, highlighting the need of using turbulence models.

8.4. Burgers equation in Fourier space

The Navier-Stokes equations are a good mathematical model to solve the motion of a turbulent fluid by means of the Fractional Step Method. The first step before doing so is to study the Burgers equation in a one-dimensional domain, which is a simplified model that shares many of the aspects with the Navier-Stokes equations [22]. In the physical space, the Burgers equation is defined in (125) for a velocity field $u(x,t)$.

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} = \frac{1}{Re}\frac{\partial^2 u}{\partial x^2} + f \tag{125}$$

It consists on the Navier-Stokes momentum equation in 1-D without pressure field nor mass conservation equation, with a source term f. This equation is going to be solved in the Fourier space, on an interval Ω with periodic boundary conditions. To transform equation (125) from the physical to the Fourier space, the following relation is used.

$$u(x,t) = \sum_{k=-N}^{k=+N} \hat{u}_k e^{ikx} \tag{126}$$

The parameter k is the wave number, N is the total number of Fourier modes and i is the imaginary unit. The variable $\hat{u}_k(t)$ stands for the k-th Fourier coefficient of $u(x,t)$, which is a time-dependant and complex. Since $u(x,t) \in \mathbb{R}$, $\hat{u}_k$ must be equal to his complex conjugate, denoted by $\overline{(\cdot)}$.

$$\hat{u}_k = \overline{\hat{u}_{-k}} \tag{127}$$

The different terms of equation (125) can be transformed into the Fourier space separately. The transient, diffusive and source terms correspond to equations (128), (129) and (130), respectively.

$$\frac{\partial u}{\partial t} = \sum_{k=-N}^{k=+N} \frac{\partial \hat{u}_k}{\partial t} e^{ikx} \tag{128}$$

$$\frac{\partial^2 u}{\partial x^2} = \sum_{k=-N}^{k=+N} \hat{u}_k \frac{\partial^2 e^{ikx}}{\partial x^2} = \sum_{k=-N}^{k=+N} \left(-k^2 \hat{u}_k\right) e^{ikx} \tag{129}$$

$$f = \sum_{k=-N}^{k=+N} F_k e^{ikx} \tag{130}$$

In regard to the convective term (source of nonlinearity), it is given by equation (132).

$$\frac{\partial u}{\partial x} = \sum_{k=-N}^{k=+N} \hat{u}_k \frac{\partial e^{ikx}}{\partial x} = \sum_{k=-N}^{k=+N} ik\hat{u}_k e^{ikx} \tag{131}$$

$$u\frac{\partial u}{\partial x} = \sum_{p=-N}^{p=+N} \hat{u}_p e^{ipx} \sum_{q=-N}^{q=+N} iq\hat{u}_q e^{iqx} = \sum_{p=-N,\, q=-N}^{p=+N,\, q=+N} \hat{u}_p iq\hat{u}_q e^{i(p+q)x} \tag{132}$$

To sum up, the Burgers equation in Fourier space stands for

$$\sum_{k=-N}^{k=+N} \frac{\partial \hat{u}_k}{\partial t} e^{ikx} + \sum_{p=-N,\, q=-N}^{p=+N,\, q=+N} \hat{u}_p iq\hat{u}_q e^{i(p+q)x} = \frac{1}{Re} \sum_{k=-N}^{k=+N} \left(-k^2 \hat{u}_k\right) e^{ikx} + \sum_{k=-N}^{k=+N} F_k e^{ikx} \tag{133}$$

where each Fourier mode is solved separately from the others, taking into account the condition $p+q = k$. Therefore, the equation to be solved is defined in (134). Note that only the positive modes need to be solved ($k \geq 0$) since the negative ones come forthrightly from condition (127).

$$\frac{\partial \hat{u}_k}{\partial t} + \sum_{k=p+q} \hat{u}_p iq\hat{u}_q = -\frac{k^2}{Re}\hat{u}_k + F_k \qquad k = 0,\ 1,\ ...,\ N \tag{134}$$

8.5. LES SPECTRAL EDDY-VISCOSITY MODEL

A Large-Eddy Simulation (LES) is a turbulence model that consists on avoiding the smallest scales of the problem. There are different strategies for doing so, but in the Fourier space it is convenient to employ a spectral eddy-viscosity model. This is based on considering a k-dependent additional viscosity in order to prematurely dissipate the energy that would be dissipated on the smallest scales of the problem. In order to do so, Métais and Lesieur proposed equation (135) in 1992, regarding the additional kinematic viscosity (called turbulent viscosity) as a function of k [23].

$$\nu_t(k/k_N) = \nu_t^{+\infty} \left(\frac{E_{k_N}}{k_N}\right)^{1/2} \nu_t^* \left(\frac{k}{k_N}\right) \tag{135}$$

The variable $\nu_t^{+\infty}$ stands for

$$\nu_t^{+\infty} = 0.31 \frac{5-m}{m+1} \sqrt{3-m} C_K^{-3/2} \tag{136}$$

where m is the slope of the curve $E(k)$ in logarithmic scale, this is the exponent of k^{-m}. In the Navier-Stokes equations, it takes the value of 5/3, but in the Burgers equation it stands for $m = 2$. The parameter C_K is the Kolmogorov constant for the one-dimensional Burgers equation, which is equivalent to $C_K \approx 0.4523$.

The variable k_N is the cutoff wave number, which corresponds to a cutoff energy level of E_{k_N}. This can be computed as $E_{k_N} = \hat{u}_{k_N}\overline{\hat{u}_{k_N}}$, which is a real number by definition. The right part of the RHS of (135) is defined in equation (137), which is equal to 1 for small values of k/k_N.

$$\nu_t^* \left(\frac{k}{k_N} \right) = 1 + 34.5e^{-3.03\left(\frac{k_N}{k}\right)} \tag{137}$$

Finally, the only modification in order to apply this model is to consider an effective kinematic viscosity, using $\nu_{eff}(k/k_N) = \nu + \nu_t(k/k_N)$ instead of ν, which in this case is equal to the inverse of the Reynolds number ($\nu = Re^{-1}$). Therefore, the diffusive term of the Burgers equation is conceived as $-k^2\nu_{eff}(k/k_N)\hat{u}_k$.

8.6. Resolution of the Burgers equation

8.6.1. Practical considerations

Equation (134) is discretized by means of a fully explicit first-order Euler scheme, yielding equation (139). For the sake of simplification, no source term is considered. The input of energy is imposed taking $\hat{u}_1 = 1$ for all instants. Furthermore, no mean flow is assumed since $k = 0$ has no interactions with other modes ($\hat{u}_0 = 0$).

$$\int_{t^n}^{t^{n+1}} \frac{\partial \hat{u}_k}{\partial t} dt = -\int_{t^n}^{t^{n+1}} \left(\frac{k^2}{Re}\hat{u}_k + \sum_{k=p+q} \hat{u}_p iq \hat{u}_q \right) dt \tag{138}$$

$$\hat{u}_k^{n+1} = \hat{u}_k^n - \Delta t \left(\frac{k^2}{Re}\hat{u}_k^n + \sum_{k=p+q} \hat{u}_p^n iq \hat{u}_q^n \right) \tag{139}$$

Note that the Reynolds number of the diffusive term can be computed as explained in Section 8.5. Regarding the time step, it is set with a CFL-like condition, as the temporal discretization is not stable for all the values of Δt. The parameter C_1 is a constant coefficient that can be modified, taking into account that it must be lower than one ($C_1 \leq 1$).

$$\Delta t < C_1 \frac{Re}{N^2} \tag{140}$$

In regard to the convective term, for each value of $k \leq N$ there are several values of p and q that verify the sum $k = p+q$. These combinations of p and q are called triadic interactions between modes, which are represented in Figure 41. As aforementioned, only the positive modes shall be computed $k > 0$, being $k = 0$ the mean flow, assumed as zero. There may be the case when negative values of p and q are found, which must be computed by means of relation (127).

The algorithm is not deeply detailed due to its relative simplicity. It consists on computing equation (139) from mode $k = 1$ until $k = N$ for each time step, imposing the condition $\hat{u}_1 = 1$, considering the triadic interactions for the convective term and the effective viscosity for the diffusive term. The convergence in time is evaluated with a residual, computed as $max|\hat{u}_k^{n+1} - \hat{u}_k^n|$. Finally, the cutoff energy E_{k_N} of equation (135) is computed at the instant t^n. No linear solver is needed.

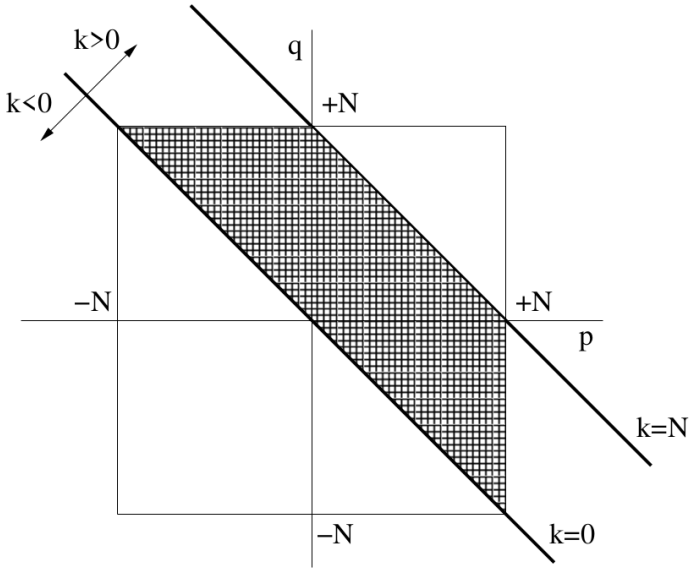


Figure 41: Illustration of all possible triadic interaction between modes [22].

8.6.2. Results

The results of the simulation for $Re = 40$ are included in Figure 42, which can be compared with the reference plot included in Figure 43. Both illustrate the energy spectrum of the steady state for an underresolved DNS simulation with $N = 20$, a LES simulation with $N = 20$ for $C_k = 0.4523$ and $C_k = 0.05$, and a DNS with $N = 100$. The steady state is found around $t \approx 3$ *sec*. The numerical parameters are chosen as $C_1 = 0.01$ for the CFL-like condition and 10^{-7} for the residual. Note that there is a mistake on the legend of Figure 43, as it refers to $C_k = 0.4523$ and not to $C_k \neq 0.4223$.

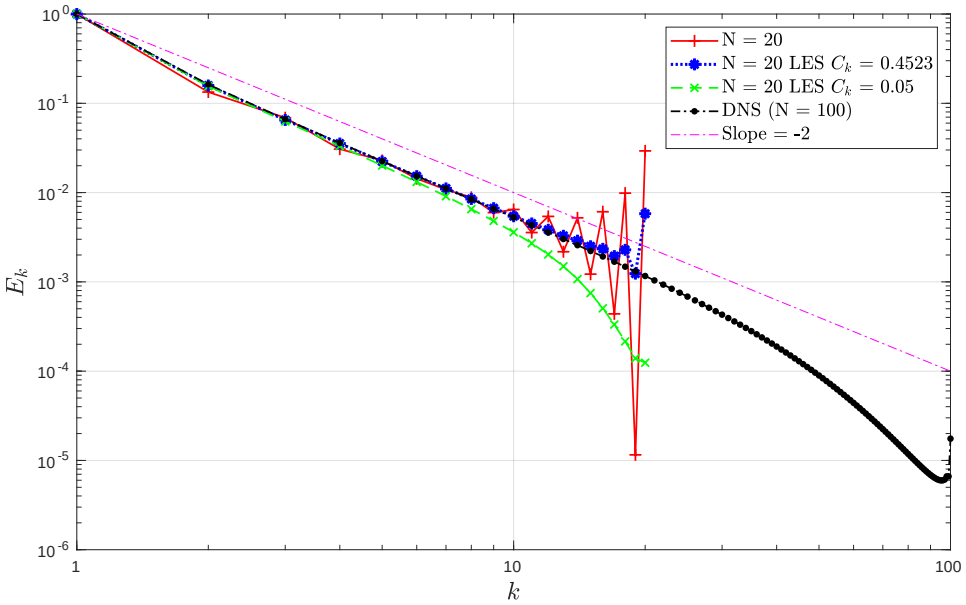


Figure 42: Burgers Equation. Results of the simulation, energy spectrum of the steady state.

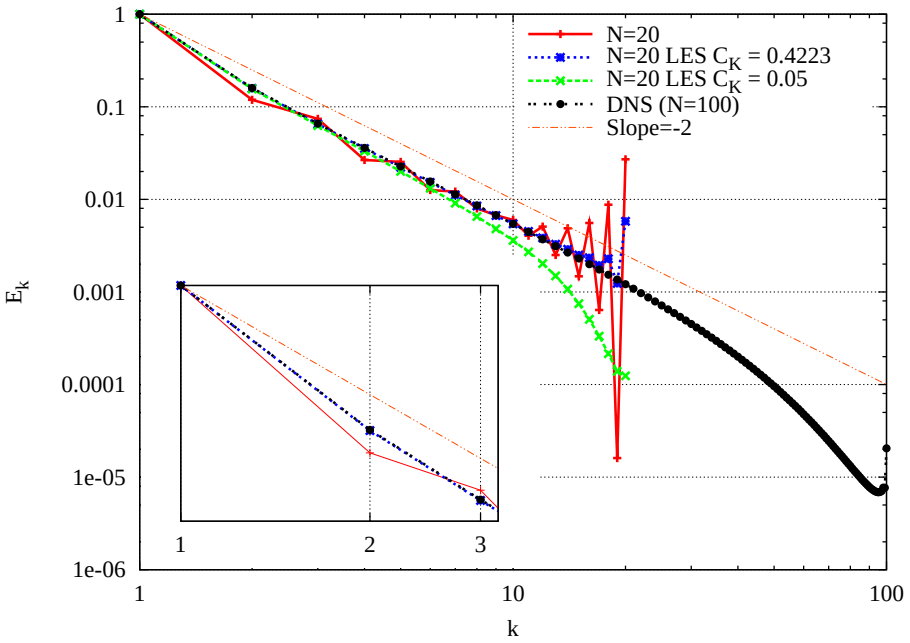


Figure 43: Burgers Equation. Reference plot given by CTTC, energy spectrum of the steady state [22].

As it can be seen, the results of the reference coincide with the ones of the simulation. For this reason, the code is considered to be verified. In regard to the results of the DNS ($N = 100$), the energy tends to decrease with a straight line with negative slope (in logarithmic scale) until $k \approx 20$, where it starts to decrease faster due to the increase of the dissipation effects. This behaviour represents the inertial range and the dissipation zones of the spectrum. A small overshoot can be seen near to $k \approx 100$, corresponding to the energy that is left to dissipate for higher modes.

When decreasing the number of simulated modes ($N = 20$), the results become highly unstable due to the lack of small scales to dissipate the remaining energy. This highlights the importance of simulating the small scales, which play a main role in the dissipation effects (despite carrying very small amounts of kinetic energy). It is possible to avoid these instabilities by using a LES turbulence model, which enables to dissipate this energy in the higher scales.

A LES simulation with $C_k = 0.4523$ is carried out for $N = 20$. As it can be seen, the results are way more stable than for the last case, as the increase on the kinematic viscosity enables to dissipate more energy. This allows to obtain better results without the necessity of simulating the smallest scales of the problem, notably reducing the simulation time.

The last simulation corresponds to a LES with $C_k = 0.05$, carried out for $N = 20$. As it can be observed, the inertial range is way shorter and the energy is dissipated on very large scales. This results are not realistic on the small scales, since the fact of decreasing the parameter C_k increases a lot the viscosity and dissipation effects, dissipating more energy than the necessary for a simulation with 20 modes. Despite this, this kind of simulation could be interesting if the intention is to simulate only the largest scales of the problem, decreasing even more the value of N.

9. The Turbulent Channel Flow Problem

9.1. Problem definition

In this section, a turbulent channel flow is solved by means of a LES simulation. The geometry and coordinate system of the present problem are defined in Figure 44.

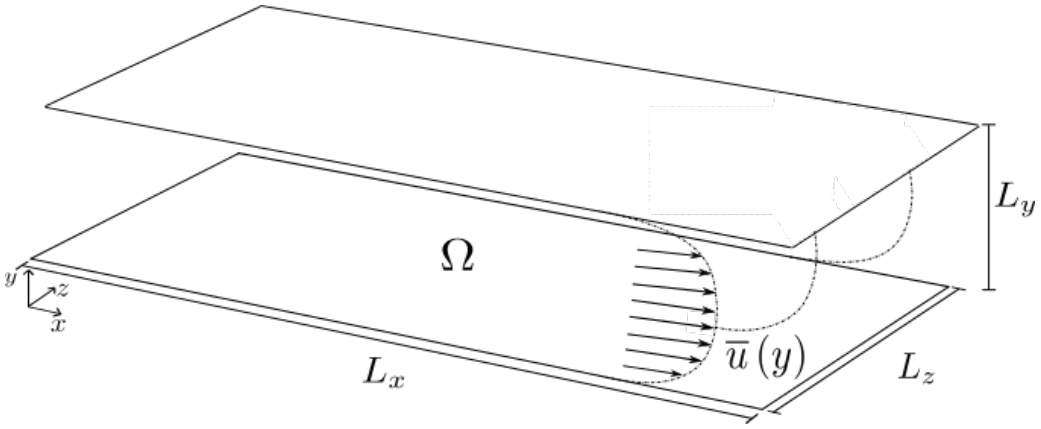


Figure 44: Illustration of the channel flow domain [24].

Some important definitions must be taken into account. First of all, the physical dimensions of the channel are described as specified in Table 12. With this in mind, a friction Reynolds number is defined as follows.

$$Re_\tau = \frac{\delta u_\tau}{\nu} \tag{141}$$

The parameter u_τ is the shear velocity, defined in equation (142). For the resolution of this problem, the shear velocity is set to $u_\tau = 1\ m/s$, the friction Reynolds number is specified and the kinematic viscosity is computed from (141).

$$u_\tau = \sqrt{\nu \left.\frac{\partial u}{\partial y}\right|_{wall}} \tag{142}$$

The dimensionless variables in wall units (used for plotting the results) are as follows.

$$u^+ = \frac{u}{u_\tau}, \qquad y^+ = \frac{y u_\tau}{\nu} \tag{143}$$

Finally, the size of the geometry is modified with Re_τ as defined in Table 12.

Re_τ	L_x	L_y	L_z
180	$4\pi\delta$	2δ	$\frac{4}{3}\pi\delta$
395	$2\pi\delta$	2δ	$\pi\delta$
590	$2\pi\delta$	2δ	$\pi\delta$

Table 12: Channel Flow. Dimensions of the domain for different values of Re_τ [25].

9.2. Practical considerations

9.2.1. Extension to 3-D

This problem is solved in a three-dimensional mesh in order to take into account the actual behaviour of the turbulent flow. In order to do so, the Fractional Step Method algorithm (see Section 4) must be extended to a three-dimensional domain. The main and only difference is the addition of a staggered mesh in the z-direction, which yields to the following changes on the algorithm.

- The first step is to evaluate $\boldsymbol{R}(\mathbf{u})$ on each node. For the computation of equation (71), the third dimension must be taken into account by adding top and bottom nodes to the calculation. An example of this is given in equation (144), which stands for the component of $\boldsymbol{R}(\mathbf{u})$ in the z-direction. The rest of dimensions are completely analogous.

$$R(w)\Omega_{zP} \approx -\frac{1}{\rho}\left[\dot{m}_e w_e - \dot{m}_w w_w + \dot{m}_n w_n - \dot{m}_s w_s + \dot{m}_t w_t - \dot{m}_b w_b\right] + \tag{144}$$
$$\frac{1}{Re}\left[\frac{w_E - w_P}{d_{EP}}A_e - \frac{w_P - w_W}{d_{WP}}A_w + \frac{w_N - w_P}{d_{NP}}A_n - \frac{w_P - w_S}{d_{SP}}A_s + \frac{w_T - w_P}{d_{TP}}A_t - \frac{w_P - w_B}{d_{BP}}A_b\right]$$

- The second step is to compute the predictor velocity field $\mathbf{u}^p$ of equation (76). In order to do so, the new velocity field in the z-direction is considered as shown in equation (145).

$$w^p = w^n + \Delta t\left(\frac{3}{2}R(w^n) - \frac{1}{2}R(w^{n-1})\right) \tag{145}$$

- The third step is to evaluate $\nabla \cdot \mathbf{u}^p$ and solve the Poisson equation (78). Two new discretization coefficients arise from the fact of considering top and bottom nodes in the third dimension. Thus, the new equation to solve with a linear solver is shown in (146), where the corresponding discretization coefficients are defined in equation (147).

$$a_P p_P^{n+1} = a_E p_E^{n+1} + a_W p_W^{n+1} + a_N p_N^{n+1} + a_S p_S^{n+1} + a_T p_T^{n+1} + a_B p_B^{n+1} + b_P \tag{146}$$

$$\begin{cases} a_E = \frac{A_e}{d_{EP}} \qquad a_W = \frac{A_w}{d_{WP}} \\ a_N = \frac{A_n}{d_{NP}} \qquad a_S = \frac{A_s}{d_{SP}} \\ a_T = \frac{A_t}{d_{TP}} \qquad a_B = \frac{A_b}{d_{BP}} \\ a_P = a_E + a_W + a_N + a_S + a_T + a_B \\ b_P = -\frac{1}{\Delta t}\left[u_e^p A_e - u_w^p A_w + v_n^p A_n - v_s^p A_s + w_t^p A_t - w_b^p A_b\right]. \end{cases} \tag{147}$$

- The last step is to compute the new velocity field $\mathbf{u}^{n+1}$ by the use of equation (75). The third dimension is added with equation (148), where the subscripts A and B are analogous to the ones defined in Figure 19 but for the staggered-z mesh.

$$w_P^{n+1} = w_P^p - \Delta t\frac{p_B^{n+1} - p_A^{n+1}}{d_{BA}} \tag{148}$$

9.2.2. LES Smagorinsky model

In order to reduce the computational effort needed for the simulation, a LES Smagorinsky turbulence model is programmed. This model adds an additional turbulence viscosity to the problem in order to dissipate the subgrid-scale eddies not considered due to the lack of mesh refinement. A spatial filtering $\overline{(\cdot)}$ is applied to the Navier-Stokes equations (55) and (56) in order to obtain equations (149) and (150).

$$\frac{\partial \overline{\mathbf{u}}}{\partial t} + (\overline{\mathbf{u}} \cdot \nabla)\overline{\mathbf{u}} = \frac{1}{Re}\Delta\overline{\mathbf{u}} - \nabla\overline{p} - \nabla \cdot \boldsymbol{\tau}(\overline{\mathbf{u}}) \tag{149}$$

$$\nabla \cdot \overline{\mathbf{u}} = 0 \tag{150}$$

The variable $\boldsymbol{\tau}(\overline{\mathbf{u}})$ is the subgrid stress tensor which aims to approximate the effect of the under-resolved scales [26]. There are different approaches for this tensor, despite most of the LES methods use the eddy-viscosity assumption defined in (151).

$$\boldsymbol{\tau}(\overline{\mathbf{u}}) = -2\nu_t \mathbf{S}(\overline{\mathbf{u}}) \tag{151}$$

Note that the different eddy-viscosity models bring different approaches for the turbulent viscosity ν_t. The Smagorinsky model is the one studied in this thesis, which stands for the approach given in equation (152).

$$\nu_t = (C_s\delta)^2 ||\mathbf{S}(\overline{\mathbf{u}})|| \tag{152}$$

On the one hand, the parameter C_s is the Smagorinsky constant[11], which is between $C_s \in [0.10,\ 0.20]$. The variable δ is the filter length, computed locally on each element as $\delta = (\Delta x \Delta y \Delta z)^{1/3}$. On the other hand, the strain tensor of the filtered velocity field $\mathbf{S}(\overline{\mathbf{u}})$ is defined as follows.

$$\mathbf{S}(\overline{\mathbf{u}}) = \frac{1}{2}\left(\mathbf{u} + \mathbf{u}^T\right) = \begin{bmatrix} \frac{\partial u}{\partial x} & \frac{1}{2}\left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right) & \frac{1}{2}\left(\frac{\partial u}{\partial z} + \frac{\partial w}{\partial x}\right) \\ \frac{1}{2}\left(\frac{\partial v}{\partial x} + \frac{\partial u}{\partial y}\right) & \frac{\partial v}{\partial y} & \frac{1}{2}\left(\frac{\partial v}{\partial z} + \frac{\partial w}{\partial y}\right) \\ \frac{1}{2}\left(\frac{\partial w}{\partial x} + \frac{\partial u}{\partial z}\right) & \frac{1}{2}\left(\frac{\partial w}{\partial y} + \frac{\partial v}{\partial z}\right) & \frac{\partial w}{\partial z} \end{bmatrix} \tag{153}$$

The magnitude of the strain tensor appears in equation (152). For a symmetric tensor of order n in Einstein notation, this operation is defined in equation (154).

$$||\mathbf{S}(\overline{\mathbf{u}})|| = \sqrt{\sum_{i,j=1}^{n} |S_{ij}|^2} \tag{154}$$

These changes are introduced into the algorithm by means of modifying equation (71), which takes the form defined in (155). As for the discretization of $\nabla \cdot \boldsymbol{\tau}(\overline{\mathbf{u}})$, the first step is to compute the turbulent viscosity locally on each node of the main mesh. Then, the subgrid stress tensor is computed on each node (three components for dimension), and the divergence operator is applied on these components.

$$\boldsymbol{R}(\overline{\mathbf{u}}) = \frac{1}{Re}\Delta\overline{\mathbf{u}} - (\overline{\mathbf{u}} \cdot \nabla)\overline{\mathbf{u}} - \nabla \cdot \boldsymbol{\tau}(\overline{\mathbf{u}}) \tag{155}$$

[11] The recommended value of C_s for a low-Reynolds turbulent Channel Flow is $C_s = 0.10$ [24].

9.2.3. Resolution method

The analytical solution of a laminar Poiseuille flow in a two-dimensional Channel Flow gives the following distribution for the velocity field, where $\tau_w = \rho\nu \left.\frac{\partial u}{\partial y}\right|_{wall}$ is the wall shear stress [27].

$$u(y) = \frac{\tau_w \delta}{2\rho\nu} \frac{y}{\delta} \left(2 - \frac{y}{\delta}\right) \tag{156}$$

This velocity distribution is obtained from the Navier-Stokes equations without imposing any hypotheses regarding the Reynolds number. This means that equation (156) is valid for all values of Re, despite the laminar solution will be unstable, yielding turbulent solutions when Re is high. The balance of forces in the channel can be studied by means of equation (157).

$$-\delta \frac{\partial p}{\partial x} = \tau_w \longrightarrow \frac{\partial p}{\partial x} = -u_\tau^2 \tag{157}$$

In other words, there is a balance between the shear stress forces acting on the boundary layer (trying to stop the fluid) and the pressure difference that tries to accelerate the flow. As the boundary conditions are periodic in the x and z directions, this balance must be imposed in the resolution algorithm by adding an additional forcing term that accelerates the flow until these forces are balanced. The additional forcing source term $\mathbf{f}_{b,S}$ on the right hand side of the momentum equation is defined as follows [24].

$$\mathbf{f}_{b,S} = \begin{bmatrix} u_\tau^2 \\ 0 \\ 0 \end{bmatrix} \tag{158}$$

This change is imposed in the algorithm by means of adding $\mathbf{f}_{b,S}$ to equation (71). In fact, the only difference in the code is the addition of a constant u_τ^2 into the x-component of $\boldsymbol{R}(\mathbf{u})$. This is known as Constant Gradient method, which is the resolution strategy used in the present thesis. There is an alternative way of solving the problem called Flux Adjustment method, the differences of which are noted below.

- **The Constant Gradient method**. The friction Reynolds number is specified and set to $Re_\tau = 395$ in the case of the present dissertation. As for the shear velocity, it is set to $u_\tau = 1\ m/s$. Then, the kinematic viscosity is computed with equation (141). The forcing term and the kinematic viscosity are taken into account in the calculation of $\boldsymbol{R}(\mathbf{u})$. A constant velocity field with a random perturbation is imposed as initial condition. The simulation is run until the flux stabilizes, when data is started to be sampled.

- **The Flux Adjustment method**. The additional forcing term is not constant. The mass flux is specified as an input of the problem and the forcing term is adjusted at each time step such that the correct mass flux is obtained. This method requires to know the correct flux for a given Re_τ and a way of updating the forcing term. This method is faster than the Constant Gradient in terms of waiting time to start sampling data, but relatively slower in terms of computing effort per iteration.

9.2.4. Turbulent boundary layers

A very important topic in order to obtain good physical results in turbulent simulations is the study of the refination of the mesh in the near-wall zones. Figure 45 illustrates the generic behaviour of any time-averaged turbulent boundary layer in wall units.

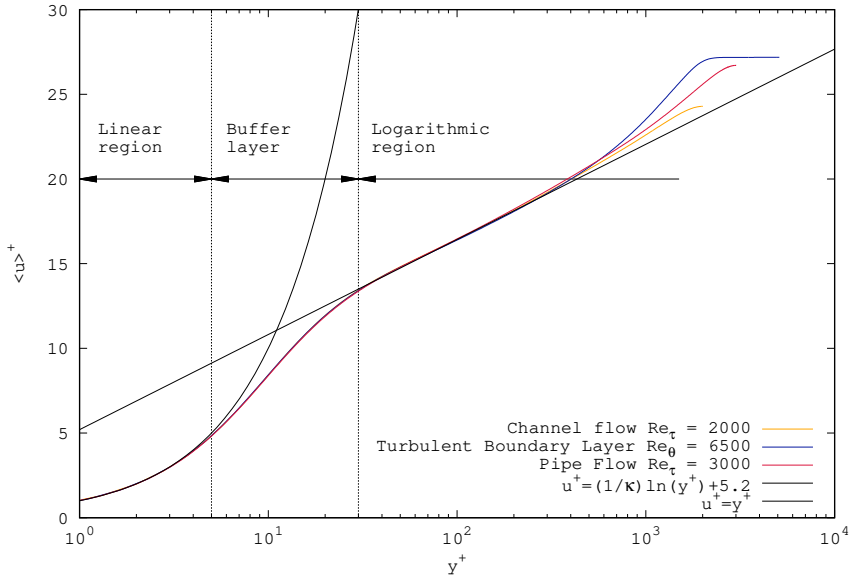


Figure 45: Channel Flow. Velocity profile of a turbulent boundary layer in wall units [28].

As it is possible to see, the velocity profile can be expressed as a function of y^+ called *law of the wall*, $u^+ = f(y^+)$. Three different regions are differentiated.

- **Linear region** located on the nearest position to the wall. It is usually called viscous sublayer, where the effects of the viscosity are the greatest. The flow has a linear behaviour defined by $u^+ = y^+$ until $y^+ < 5$. The gradient of the velocity is very steep.
- **Buffer layer** located between $y^+ > 6$ and $y^+ < 30$. This zone acts as a transition between the linear and logarithmic regions. The non-linear turbulent effects start to have a relevant role.
- **Logarithmic region** from $y^+ > 30$ until the bulk flow. The fluid is completely turbulent, it can be assumed that ν has a very small effect. It can be proved that $u^+ = \frac{1}{\kappa} \ln y^+ + B$, where $\kappa \approx 0.41$ is the von Kármán constant and $B \approx 5.2$.

As the velocity gradient on the walls is specified by the parameter u_τ, the slope of the linear region is indirectly imposed in the code. For this reason, it is very important to define a first node inside this linear region. Otherwise, the velocity of the first node will be over-estimated, yielding non-physical results in the bulk flow. This condition is critical in the development of any CFD code, imposing high restrictions in the size of the mesh.

Summarizing, a mesh for a DNS simulation must take into account two different aspects. The first of them is that the mesh of the bulk flow must be capable of computing the Kolmogorov length scale. In order to make it possible in the present dissertation, a LES turbulence model is used in order to dissipate the subgrid scales. The second aspect is the definition of the mesh in the near-wall zones, which must include at least one node inside the linear region of the boundary layer.

9.2.5. Mesh definition

For the sake of simplification, only the case with $Re_\tau = 395$ is analyzed in the present dissertation. The chosen domain has the dimensions specified in Table 12. Moreover, the dimensions of the mesh are chosen in order to obtain good results taking into account the computing effort of the computer running the simulation.

The mesh is uniform and equidistant in the x and z dimensions, with 30 and 20 elements on their respective directions. Regarding the y-direction, the position of the elements follows the hyperbolic tangent relation defined in equation (159). The variable M is the number of elements in the y-direction, 45 in this case. The parameter γ is set to $\gamma = 1$ to have a symmetric mesh, and k is the indexing of the element[12].

$$y_k = y_0 + \delta \left[1 + \frac{\tanh\left(\frac{2\gamma k}{M} - 1\right)}{\tanh\left(\gamma\right)} \right] \tag{159}$$

It is important to note that there must be enough elements in the y-direction in order to properly define the near-wall zones. Using the mentioned hyperbolic function, the first node of the main mesh is located at $y = 12.67\ mm$ from the wall. For $Re_\tau = 395$ and $u_\tau = 1\ m/s$, a kinematic viscosity of $\nu = 2.532 \cdot 10^{-3}\ m^2/s$ is obtained. Using equation (143), the position of the first node in wall units is computed, yielding $y^+ = 5.01$.

As the linear region of the turbulent boundary layer is found for $y^+ < 5$, the position of the first node ($y^+ = 5.01$) of the employed mesh is just at the limit to start giving relatively good results. Ideally, the position of the first node when using a turbulence model should be even closer to the wall ($y^+ \approx 1$) to properly define the mesh, despite this is not possible due to the high computing cost of the simulation. The LES Smagorinsky model is expected to increase the kinematic viscosity in the boundary layer, slightly under-estimating the solution.

To summarize, the main mesh consists on $27,000$ hexahedral elements. The element distribution in the $x-y-z$ directions is $30 \times 45 \times 20$. Figure 46 illustrates an XY slice of the mesh, which is repeated 20 times in the z-direction.

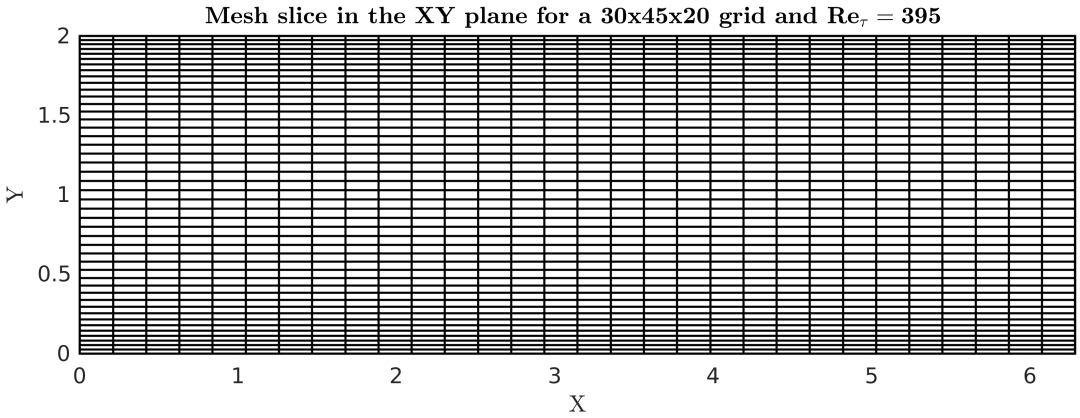


Figure 46: Channel Flow. Representation of a main mesh slice in the XY plane.

[12]Note that the first element is located at $k = 0$ in C++.

9.2.6. Boundary conditions and remarks on the algorithm

All the velocity boundary conditions are imposed in the calculation of the new velocity field. In the y-direction, no-slip conditions are imposed on the bottom and top walls to the three components of the velocity field. In the x and z directions, the boundary conditions are periodic. This means that the velocity of the fluid leaving the domain at t^n is imposed as the velocity entering the domain at t^{n+1}. By applying this kind of boundary condition, the domain acts as if it was infinite in the x and z directions, making the flow capable of becoming fully-developed in terms of turbulence effects.

The pressure boundary conditions are applied in the calculation of the discretization coefficients of the Poisson equation. These are zero-gradient Neumann boundary conditions ($\frac{\partial p}{\partial n} = 0$) on the walls, entrance and z-direction. On the outlet, the boundary condition is Dirichlet and the pressure is set to $p = 0$.

Finally, the algorithm is completely analogous to the Lid Driven Cavity and Square Cylinder problems. For this reason, no further explanation is included and only the differences are commented. The main differences are the ones described above: the extension to 3-D, the addition of the subgrid stress tensor on the calculation of $\boldsymbol{R}(\mathbf{u})$, the additional forcing term imposed in the x-component of $R(u)$ and the particularities of the periodic boundary conditions.

9.3. Results

In this section, the results obtained from simulating the Channel Flow at $Re_\tau = 395$ with a LES Smagorinsky model with $C_s = 0.10$ are discussed. A 2-D instantaneous plot of the velocity magnitude and turbulent kinematic viscosity for the fully-developed turbulent flow is included in Figure 47.

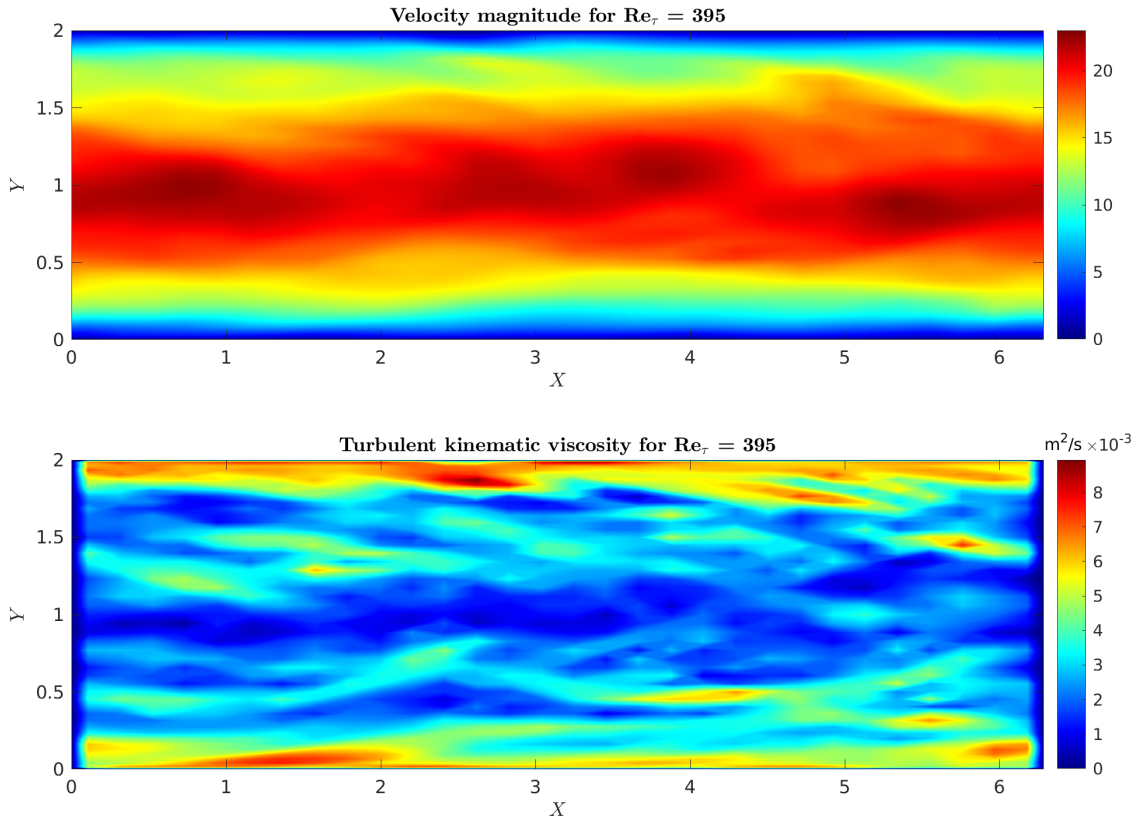


Figure 47: Channel Flow. Velocity magnitude and turbulent kinematic viscosity for $Re_\tau = 395$. See VIDEO.

As mentioned above, the simulation is run until a statistically-steady regime is found, where the turbulent flow is fully-developed. Note that a VIDEO is attached to help visualizing the transient evolution of the turbulent flow. It can be appreciated how the LES turbulence model acts on the zones where the velocity gradient is greater. The boundary layer zones are the positions where the turbulent kinematic viscosity is the greatest, as the flow changes its velocity more rapidly. Otherwise, the bulk flow is almost not affected by the turbulence model, as expected. Note that as $u_\tau = 1\ m/s$, the magnitude of the velocity field is the same for the dimensionless and primitive variables.

The velocity field can be averaged in time in order to find the time-averaged solution, which shall be symmetric in the y-direction having its maximum around the middle-height of the channel ($y = \delta$). The kinetic energy of the flow per unit of mass $e_k = \frac{<|\mathbf{u}|>^2}{2}$ is computed and shown in Figure 48, where a completely symmetric behaviour is found.

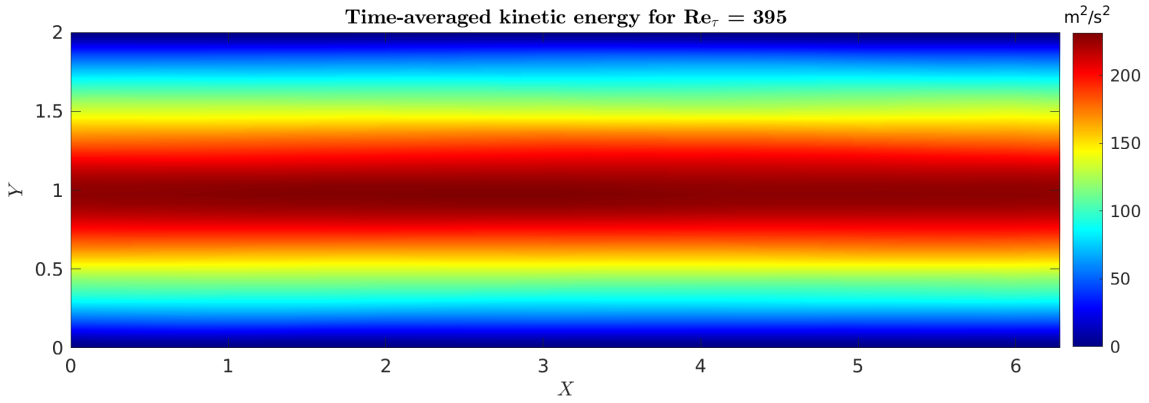


Figure 48: Channel Flow. Time-averaged kinetic energy for $Re_\tau = 395$.

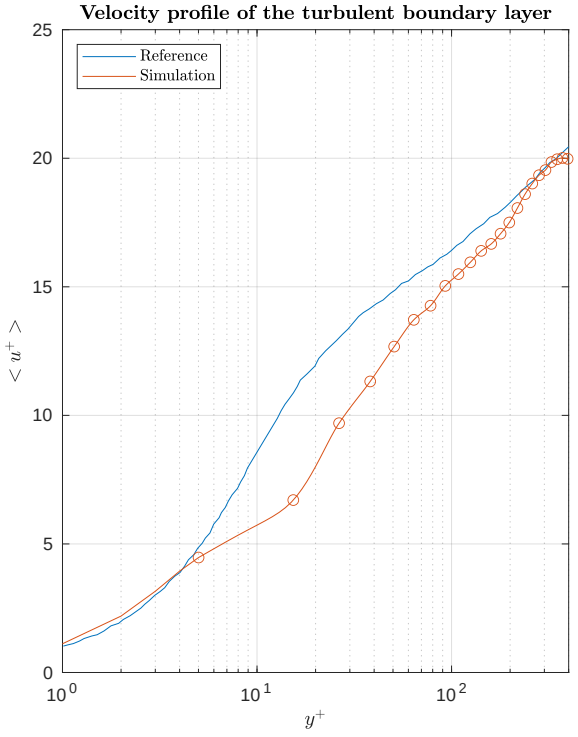


Figure 49: Channel Flow. Turbulent boundary layer comparison with reference [25].

Finally, the time-averaged velocity field profile in the boundary layer is compared with reference values in order to verify the code. As it can be seen in Figure 49, there are important differences between the simulation and the reference outside the viscous sublayer of the boundary layer.

Regarding the velocity of the first node, it is indirectly imposed in the algorithm. The behaviour between the wall and the first node must be linear. For this reason, a good result in the velocity of the first node is obtained, as it is located just at the limit of the linear region.

In regard to the buffer layer, as the size of the mesh is not refined enough, the LES Smagorinsky model yields relatively high values on the value of ν_τ. For this reason, the solution is under-estimated from the second node to the bulk flow. As y^+ increases and the logarithmic region is reached, the turbulent kinematic viscosity decreases and the solution tends to be closer to the reference.

There are two ways of avoiding the obtained behaviour: decreasing the value of the constant C_s or increasing the size of the mesh. However, the first of the options can not be done as the mesh is too coarse to dissipate the subgrid scales if the constant C_s is reduced. This would yield an unstable solution where the velocity field oscillates in the regions where turbulence is more intense. Regarding the second option, the mesh can not be increased due to the high computational cost limitations of the author of the present thesis.

An ideal mesh size would have the first node around $y^+ \approx 1$. In order to obtain this, around 200 elements would be needed in the y-direction. The x and z directions do not need such a high mesh refinition, despite the x-direction is twice larger than the z-direction. For this reason, 200 more elements in the x-direction and 100 elements in the z-direction are proposed. This would give a mesh of 4M elements, probably good enough to perform a DNS simulation [25].

In the present case, the LES Smagorinsky model could radically reduce the mentioned mesh size. For instance, having a first node around $y^+ \approx 2.5$ would entail a mesh with $90 \times 90 \times 45$ elements. This means having twice the number of elements in the y-direction than the simulation of the present dissertation, yielding a total of $364,500$ elements (13.5 times larger than the employed mesh). By doing this, the mesh size would probably be large enough to obtain results much closer to the reference with the help of the LES model.

Two new lines of research arise from the fact of not having enough computational power (besides the implementation of different turbulence models). The first of them is the field of High Performance Computing (HPC), which uses parallelization techniques to perform simulations in supercomputers. The second of them is the study of a more efficient linear solver, such as an iterative TDMA or a Conjugate Gradient. These ideas are furtherly explained in Section 10.2.

10. Conclusions

In this section, the work done in the present dissertation is concluded. The results obtained during the realization of the simulations are summarized. A set of recommendations for future work is included, taking into account its hypothetical temporal aspects. Furthermore, a summary of the budget and the environmental impact of the thesis are presented.

10.1. Summary of the results

The first step of the realization of this work has been the fully understanding of the theory regarding the Navier-Stokes equations. They are presented, explained and deduced from their integral to their differential form, by means of the application of the Reynolds transport theorem over a differential control volume. The hypotheses made in the deduction of the final form of the equations used for the rest of the work are deeply explained and analyzed.

The next step has been the analysis and resolution of a generic convection-diffusion equation. In order to do so, the equations are firstly discretized in space and time, analyzing some of the corresponding convective and temporal schemes used along the work. Then, a first case of pure diffusion applied to a conduction heat transfer problem in two dimensions is solved, and the results properly analyzed in terms of the numerical parameters and physical meaning. A second case consisting on the resolution of a generic convection-diffusion equation is studied for a given domain and velocity field. The results are analyzed and understood for different ratios of convective-diffusive effects.

Afterwards, the incompressible Navier-Stokes equations themselves are solved for different laminar problems, considering the coupling between the continuity and momentum equations by using the Fractional Step Method. This method predicts a fictitious velocity field, subsequently projected into a divergence-free plane by means of the Helmholtz-Hodge decomposition theorem, which enables to split the Navier-Stokes equations into a divergence-free velocity field and the gradient of the pressure field.

After introducing the theory regarding the mentioned method, an algorithm is programmed and three laminar problems in two dimensions are solved. The first of them is the Lid Driven Cavity problem, which consists on a square cavity with one wall in motion solved for different Reynolds numbers. The second problem is the Differentially Heated Cavity, consisting on a square cavity with vertical acceleration. The Boussinesq term is considered for the mass forces, and the energy equation is computed in order to obtain the temperature field for different Rayleigh numbers. The third problem consists on an external flow entering a domain with a square cylinder of very large depth within, solved for different Reynolds numbers. The separation of the boundary layer over the walls of this cylinder produces a dynamic Von Kármán vortex street.

To finalize the study, an introduction to turbulence is carried out. The origin and behaviour of a turbulence flow is studied, introducing the concepts of chaos, determinism and nonlinearity. The role of the convective and diffusive terms on different length scales is also studied, understanding the Kolmogorov hypothesis and the concept of energy cascade. An introduction to the concept of turbulence model is carried out, making emphasis in the LES spectral eddy-viscosity model. Furthermore, the Burgers equation is solved in a Fourier space, comparing the results for both DNS and LES simulations. Finally, all these concepts are applied in the resolution of a three-dimensional turbulent Channel Flow problem with the help of a LES Smagorinsky turbulence model, obtaining limited good results due to restrictions on the computational power.

10.2. Recommendations

The work done in the present dissertation is very far away from the current state of the art regarding the computational resolution of the Navier-Stokes equations. Modern solvers are capable of solving complex turbulent cases in three dimensions, the input is more generalized and the efficiency of the codes is much better. Furthermore, the simulations can be run in high performance supercomputers, as they can be parallelized in order to reduce the computing time drastically.

For these reasons, it is clear that the next steps to continue developing the code are the implementation of a more efficient linear solver, the study of its parallelization and the generalization of the input. This will need a previous debug, in order to revise the current state of the solver and amend minor errors. These changes could be verified by solving a high-referenced problem, such as simulating again the Channel Flow in order to improve the results obtained in the present dissertation.

For instance, the linear solver could be improved by changing the current Gauss-Seidel to a Conjugate Gradient solver [29]. This second solver has the great advantage that can be parallelized, yielding the task of parallelizing the code by means of the C++ MPI[13] library. Afterwards, the study of other turbulence models can be done, such as a dynamic LES Smagorinsky or a RANS[14] turbulence models. Moreover, the code could be run in a supercomputer such as the Barcelona Supercomputing Center (BSC) [30], which has many collaboration agreements with the UPC.

As aforementioned, the input of the code could be generalized for more complex geometries and mesh-generation strategies. The idea is to make the code able to understand complex mesh data files created by preprocessing comercial software. The post-processing strategy could also be improved, as specialized software such as ParaView could be used instead of MATLAB. A more complex problem could be approached, such as the simulation of a subsonic aerodynamic airfoil at relatively high Reynolds numbers. All these ideas are the subject of future research, which could be the goal of a master's thesis dissertation.

10.3. Future research

In this section, the recommended work mentioned above is to be detailed and planned in several tasks. Each working week corresponds to an average of 20 hours of work.

- Task 1: Debug of the current code and amending of minor errors. Estimated time: 2 weeks.
- Task 2: Study and implementation of a Conjugate Gradient solver. Estimated time: 5 weeks.
- Task 3: Parallelization of the linear solver and the rest of the code. Estimated time: 5 weeks.
- Task 4: Study and implementation of other turbulence models, such as a dynamic LES Smagorinsky or RANS. Estimated time: 5 weeks.
- Task 5: Improving the pre-processing and post-processing strategies. Estimated time: 4 weeks.
- Task 6: Resolution of a turbulent Channel Flow problem and comparison with the results obtained in the present thesis. Estimated time: 5 weeks.
- Task 7: Resolution of an aeronautics-related problem, such as the simulation of a turbulent flow around an aerodynamic airfoil at relatively high Reynolds numbers. Hypothetically, possible collaboration with the BSC. Estimated time: 6 weeks.

This could be object of an hypothetical master's thesis dissertation based on the extension of the present final degree project. There are at least 32 weeks proposed for this future project and therefore, 640 working hours.

[13] Message Passing Interface. Standardized and portable message-passing standard designed to function on parallel computing architectures.
[14] Reynolds-Averaged Navier-Stokes are turbulence models based on filtering the Navier-Stokes equations in time.

10.4. Budget and environmental impact

To finalize the study, a summary of the budget including the costs of developing this thesis is included in Table 13. The direct costs correspond to the engineering work, while the indirect costs to the computer, energy usage and software licenses. A more detailed budget is attached alongside the report in a file named "Budget". As it can be seen, a total cost of **6831€** is estimated.

Direct cost	6200€
Indirect cost	631€
Total cost	6831€

Table 13: Summary of the costs of the thesis.

Furthermore, the environmental impact of the work is questioned. On the one hand, the aim of CFD simulations is to avoid expending vasts quantities of money in expensive experimental setups, such as wind tunnels or prototypes. As a consequence, avoiding this manufacturing also saves vasts quantities of raw materials, fossil fuels and energy. From this point of view, the present thesis is environmentally concerned, as its main objective is to generate knowledge in the field of CFD simulations. To create experienced professionals in engineering simulations is of vital importance to reduce the environmental impact of engineering projects.

On the other hand, the indirect costs of this study (detailed in the budged) have an environmental impact in terms of carbon footprint and energy usage. The laptop used for developing the thesis has been an ASUS SonicMaster X541. It carries a memory of 8GB DDR4, Graphics 620, processor i7-7500U and power input of 65 W. A total of 1240 hours of computer usage are estimated, with the half of them (620 hours) corresponding to passive engineering simulations and the other half corresponding to active engineering work. This yields an estimated total energy usage of $65\ W \cdot 1240\ h \cdot \frac{1\ kW}{10^3\ W} = 80.6\ kWh$.

The environmental costs of the computing simulations can be calculated as detailed in the following paper [31], which brings an online tool to calculate the carbon footprint and energy needed corresponding to the simulation hours as a function of the computer specifications. A total of 60.9 kWh and 13.4 kg of CO_2 are estimated for the 620 hours of simulations, which corresponds to the 76% of all the energy usage of the thesis.

Finally, the manufacturing of the computer itself has also several environmental implications. A study of the United Nations [32] stated that the costs of manufacturing a desktop computer requires at least 10 times its weight in fossil fuels and chemicals. As the computers become more powerful and smaller, their environmental impact increases. The cutting edge technological nature of computer manufacturing makes it energy-intensive and therefore important to climate change and the consumption of fossil fuel resources.

References

[1] CTTC. "Heat and Mass Transfer Technological Center - UPC". In: (2022). (Accessed on 18/02/2022). URL: https://www.cttc.upc.edu/.

[2] Essam E. Khalil. "CFD Letters. CFD History and Applications". In: (2012). (Accessed on 19/02/2022). URL: https://www.akademiabaru.com/doc/CFDLV4_N2_P43_46.pdf.

[3] Irving Shames. *Mecánica de fluidos*. Santafé de Bogota: McGraw-Hill, 1995. ISBN: 958-600-246-2.

[4] Sauro Succi. "Lecture 10: Finite Volumes". In: (2015). (Accessed on 14/03/2022). URL: https://projects.iq.harvard.edu/files/ac274_2015/files/lecture9.pdf.

[5] Suhas Patankar. *Numerical heat transfer and fluid flow*. New York London: Routledge, 1980. ISBN: 9781315275130.

[6] CTTC - Polytechnic University of Catalonia. "A Two-dimensional Transient Conduction Problem". In: (), pp. 1–3.

[7] Jonathan M. Blackledget. "Chapter 9 - Iterative Methods of Solution". In: *Digital Signal Processing (Second Edition)*. Ed. by Jonathan M. Blackledget. Second Edition. Woodhead Publishing Series in Electronic and Optical Materials. Woodhead Publishing, 2006, pp. 237–254. ISBN: 978-1-904275-26-8. DOI: https://doi.org/10.1533/9780857099457.2.237. URL: https://www.sciencedirect.com/science/article/pii/B9781904275268500105.

[8] CTTC - Polytechnic University of Catalonia. "A Two-dimensional Steady Convection-Diffusion Equation: the Smith-Hutton problem". In: (), pp. 1–3.

[9] Harsh Bhatia et al. "The Helmholtz-Hodge Decomposition-A Survey". In: *IEEE Transactions on Visualization and Computer Graphics* 19 (Nov. 2012), pp. 1386–. DOI: 10.1109/TVCG.2012.316.

[10] CTTC - Polytechnic University of Catalonia. "Introduction to the Fractional Step Method". In: (), pp. 1–10.

[11] Alexandre Joel Chorin. "Numerical solution of the Navier-Stokes equations". In: *Mathematics of computation* 22.104 (1968), pp. 745–762.

[12] CTTC - Polytechnic University of Catalonia. "Lecture notes. Course on Numerical Methods in Heat Transfer and Fluid Dynamics. Fractional Step Method - Staggered and Collocated Meshes". In: (2022), pp. 11, 17, 23.

[13] S. Mirbagheri et al. "Simulation of Liquid Flow Permeability for Dendritic Structures during Solidification Process". In: July 2011. ISBN: 978-953-307-169-5. DOI: 10.5772/19647.

[14] U. Ghia, K. N. Ghia, and C. T. Shin. "High-Re Solutions for Incompressible Flow Using the Navier-Stokes Equations and a Multigrid Method". In: *Journal of Computational Physics* 48.3 (Dec. 1982), pp. 387–411. DOI: 10.1016/0021-9991(82)90058-4.

[15] G. De Vahl Davis. "Natural convection of air in a square cavity: A bench mark numerical solution". In: *International Journal for Numerical Methods in Fluids* 3.3 (1983), pp. 249–264. DOI: https://doi.org/10.1002/fld.1650030305. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/fld.1650030305. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/fld.1650030305.

[16] Taha Khademinejad, Pouyan Talebizadeh Sardari, and Hassan Rahimzadeh. "Numerical Study of Unsteady Flow around a Square Cylinder in Compare with Circular Cylinder". In: Feb. 2015.

[17] John Anderson. *Fundamentals of aerodynamics*. New York, NY: McGraw Hill Education, 2017. ISBN: 978-1-259-12991-9.

[18] Kanchan M. Kelkar and Suhas V. Patankar. "Numerical prediction of vortex shedding behind a square cylinder". In: *International Journal for Numerical Methods in Fluids* 14.3 (Feb. 1992), pp. 327–341. DOI: 10.1002/fld.1650140306.

[19] Michael Breuer et al. "Accurate computations of the laminar flow past a square cylinder based on two different methods : lattice-Boltzmann and finite-volume". In: *International Journal of Heat and Fluid Flow* 21 (2000), pp. 186–196.

[20] F. Xavier Trias. "Lecture notes. Analysis of thermal and fluid dynamics issues in industrial and aeronautical systems and equipment - Turbulence". In: (2022), pp. 1–19.

[21] L. C. Berselli. *Mathematics of large eddy simulation of turbulent flows*. Berlin New York: Springer, 2006. ISBN: 978-3-540-26317-3.

[22] CTTC - Polytechnic University of Catalonia. "Burgers equation in Fourier space". In: (Nov. 2014), pp. 1–8.

[23] Olivier Métais and Marcel Lesieur. "Spectral large-eddy simulation of isotropic and stably stratified turbulence". In: *Journal of Fluid Mechanics* 239 (1992), pp. 157–194. DOI: 10.1017/S0022112092004361.

[24] Joakim BO. "Implementation and assessment of subgrid-scale models for large eddy simulations of incompressible turbulent flows". In: *MSc Thesis at the Faculty of Mathematics and Natural Sciences, University of Oslo* (Nov. 2015).

[25] Robert D Moser, John Kim, and Nagi N Mansour. "Direct numerical simulation of turbulent channel flow up to Re_τ= 590". In: *Physics of fluids* 11.4 (1999), pp. 943–945.

[26] CTTC - Polytechnic University of Catalonia. "Large-Eddy Simulation of turbulent incompressible flows". In: (2014), pp. 1–8.

[27] F. Xavier Trias. "Lecture notes. Analysis of thermal and fluid dynamics issues in industrial and aeronautical systems and equipment - Turbulent Boundary Layers". In: (2022), pp. 15–30.

[28] Joan Calafell. "Efficient Wall Modeling for Large Eddy Simulations of General Non-Equilibrium Wall-Bounded Flows. Doctoral Thesis at Technical University of Catalonia (UPC)". In: (May 2019). Supervisors: F. Xavier Trias and Assensi Oliva.

[29] Jonathan R Shewchuk. *An Introduction to the Conjugate Gradient Method Without the Agonizing Pain*. Tech. rep. USA, 1994.

[30] BSC-CNS. "Barcelona Supercomputing Center - Centro Nacional de Supercomputacion". In: (2022). (Accessed on 15/05/2022). URL: https://www.bsc.es/.

[31] Loïc Lannelongue, Jason Grealey, and Michael Inouye. "Green Algorithms: Quantifying the Carbon Footprint of Computation". In: *Advanced Science* 8.12 (2021), p. 2100707. DOI: https://doi.org/10.1002/advs.202100707. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/advs.202100707. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/advs.202100707.

[32] United Nations University. "UN study shows environmental consequences from ongoing boom in personal computer sales". In: (2004). (Accessed on 16/05/2022). URL: https://www.eurekalert.org/news-releases/743999.

A. Appendix I

The Appendix of the present work is divided in three parts:

- **Appendix I**: It contains the reference tables and additional plots. It is found within the report.
- **Appendix II**: It contains all the C++ code. It is attached in a file alongside the report.
- **Appendix III:** It contains the quality checklist of the work. It is attached alongside the report.

A.1. Reference data

x-position	ϕ_{outlet} $\rho/\Gamma = 10$	ϕ_{outlet} $\rho/\Gamma = 10^3$	ϕ_{outlet} $\rho/\Gamma = 10^6$
0.0	1.989	2.000	2.000
0.1	1.402	2.999	2.000
0.2	1.146	1.999	2.000
0.3	0.946	1.985	1.999
0.4	0.775	1.841	1.964
0.5	0.621	0.951	1.000
0.6	0.480	0.154	0.036
0.7	0.349	0.001	0.001
0.8	0.227	0.000	0.000
0.9	0.111	0.000	0.000
1.0	0.000	0.000	0.000

Table A.1: Smith Hutton. Reference data summary [8].

y/L	$Re = 100$	$Re = 400$	$Re = 1000$	$Re = 3200$	$Re = 5000$	$Re = 7500$	$Re = 10000$
1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
0.9766	0.8412	0.7584	0.6593	0.5324	0.4822	0.4724	0.4722
0.9688	0.7887	0.6844	0.5749	0.4830	0.4612	0.4705	0.4778
0.9609	0.7372	0.6176	0.5112	0.4655	0.4599	0.4732	0.4807
0.9531	0.6872	0.5589	0.4660	0.4610	0.4604	0.4717	0.4780
0.8516	0.2315	0.2909	0.3330	0.3468	0.3356	0.3423	0.3464
0.7344	0.0033	0.1626	0.1872	0.1979	0.2009	0.2059	0.2067
0.6172	-0.1364	0.0214	0.0570	0.0716	0.0818	0.0834	0.0834
0.5000	-0.2058	-0.1148	-0.0608	-0.0427	-0.0304	-0.0380	0.0311
0.4531	-0.2109	-0.1712	-0.1065	-0.0866	-0.0740	-0.0750	-0.0754
0.2813	-0.1566	-0.3273	-0.2781	-0.2443	-0.2286	-0.2318	-0.2319
0.1719	-0.1015	-0.2430	-0.3829	-0.3432	-0.3305	-0.3239	-0.3271
0.1016	-0.0643	-0.1461	-0.2973	-0.4193	-0.4044	-0.3832	-0.3800
0.0703	-0.0478	-0.1034	-0.2222	-0.3783	-0.4364	-0.4303	-0.4166
0.0625	-0.0419	-0.0927	-0.2020	-0.3534	-0.4290	-0.4359	-0.4254
0.0547	-0.0372	-0.0819	-0.1811	-0.3241	-0.4117	-0.4315	-0.4274
0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000

Table A.2: Lid Driven Cavity. Reference data summary for the horizontal velocity, in m/s [14].

x/L	$Re = 100$	$Re = 400$	$Re = 1000$	$Re = 3200$	$Re = 5000$	$Re = 7500$	$Re = 10000$
0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
0.0625	0.0923	0.1836	0.2749	0.3956	0.4245	0.4398	0.4398
0.0703	0.1009	0.1971	0.2901	0.4092	0.4333	0.4403	0.4373
0.0781	0.1089	0.2092	0.3035	0.4191	0.4365	0.4356	0.4312
0.0938	0.1232	0.2297	0.3263	0.4277	0.4295	0.4182	0.4149
0.1563	0.1608	0.2812	0.3710	0.3712	0.3537	0.3506	0.3507
0.2266	0.1751	0.3020	0.3308	0.2903	0.2807	0.2812	0.2800
0.2344	0.1753	0.3017	0.3224	0.2819	0.2728	0.2735	0.2722
0.5000	0.0545	0.0519	0.0253	0.0100	0.0095	0.0082	0.0083
0.8047	-0.2453	-0.3860	-0.3197	-0.3118	-0.3002	-0.3045	-0.3072
0.8594	-0.2245	-0.4499	-0.4267	-0.3740	-0.3621	-0.3621	-0.3674
0.9063	-0.1691	-0.2383	-0.5155	-0.4431	-0.4144	-0.4105	-0.4150
0.9453	-0.1031	-0.2285	-0.3919	-0.5405	-0.5288	-0.4859	-0.4586
0.9531	-0.0886	-0.1925	-0.3371	-0.5236	-0.5541	-0.5235	-0.4910
0.9609	-0.0739	-0.1566	-0.2767	-0.4743	-0.5507	-0.5522	-0.5299
0.9688	-0.0591	-0.1215	-0.2139	-0.3902	-0.4977	-0.5386	-0.5430
1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000

Table A.3: Lid Driven Cavity. Reference data summary for vertical velocity, in m/s [14].

Ra	$\overline{Nu}$	$\frac{y}{L}\big\vert_{x=L/2}$	u^*_{max}	$\frac{x}{L}\big\vert_{y=L/2}$	v^*_{max}
10^3	1.118	0.813	3.649	0.178	3.697
10^4	2.243	0.823	16.178	0.119	19.617
10^5	4.519	0.855	34.73	0.066	68.59
10^6	8.800	0.850	64.63	0.0379	219.36

Table A.4: Defferentially heated cavity. Reference data summary [15].

A.2. Square Cylinder. Additional plots.

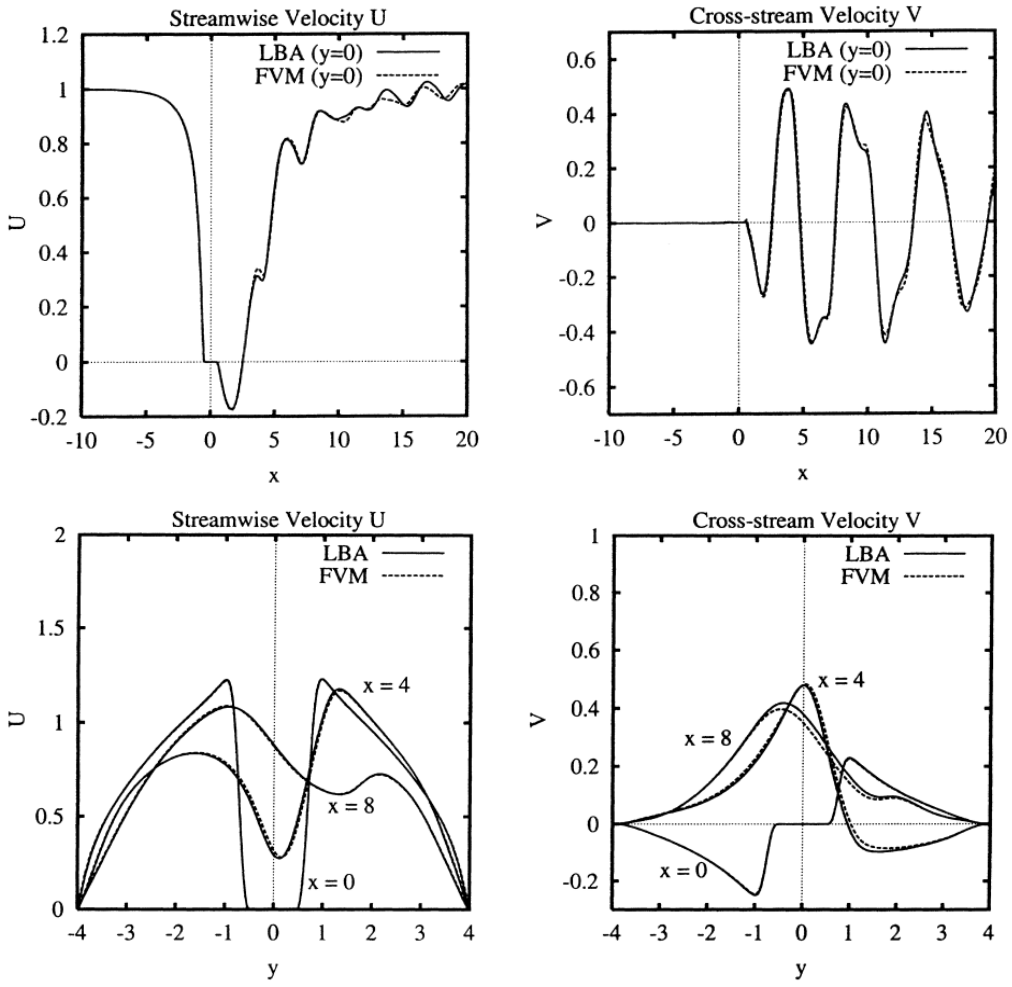


Figure A.1: Square Cylinder. Reference of the velocity distribution at different positions [19].