



UNIVERSITAT POLITÈCNICA DE CATALUNYA

BARCELONATECH

Escola Tècnica Superior d'Enginyeries  
Industrial i Aeronàutica de Terrassa

# **STUDY FOR THE NUMERICAL RESOLUTION OF CONSERVATION EQUATIONS OF MASS, MOMENTUM AND ENERGY TO BE APPLIED IN DIFFERENT ENGINEERING PROBLEMS (case 6)**

**STUDIES:** Grau en Enginyeria en Tecnologies Aeroespacials

**STUDENT:** Jordi Poblador Ibáñez

**DIRECTOR:** Asensio Oliva Llena

**CO-DIRECTOR:** Carlos David Perez Segarra

**TUTOR:** David García-Almiñana

**DELIVERY DATE:** 1st September 2015

**DOCUMENT:** Report





# **TREBALL FINAL DE GRAU**

## **REPORT**



## Acknowledgments

Before the lector starts reading the following pages, I would like to express my gratitude to all the people that made possible the elaboration of this study.

It has been a long time since I started with this adventure on October 2014, when the director, Asensio Oliva Llena, and the co-director, Carlos David Perez Segarra, gave me the opportunity to start working on this study, first of all visiting some seminars that the *Centre Tecnològic de Transferència de Calor* (CTTC) was offering about this subject. I also want to say that my interest for the computational fluid dynamics started due to the subject that I took during the second semester of the academic year 2013/2014, *Dinàmica de Gasos i Transferència de Calor i Massa*, which was given by both my director and co-director.

Furthermore, I would like to thank Daniel Crespo, father of my friend Oriol Crespo and teacher of the Universitat Politècnica de Catalunya, who helped me running some simulations when my computer was not enough powerful, and Dennis Solanki, my Swedish friend that I met during my mobility program in the Universität Stuttgart, who also helped me letting me use the computers of the university when my computer was also not enough powerful to run the simulations in a reasonable time interval.

I don't want to finish without a special gratitude to Francesc Xavier Trias Miquel, member of the CTTC, who helped me patiently with every problem I had during this study, even with the difficulty of distance and the impossibility to do personal meetings due to my mobility program this semester.

And thanks to my family, girlfriend and friends for their support during these past months.



## Contents

Acknowledgments .....	5
List of figures .....	9
List of tables .....	13
1. INTRODUCTION .....	15
1.1. Aim of the study .....	15
1.2. Scope of the study .....	15
1.3. Requirements .....	15
1.4. Justification .....	15
2. STUDY AND DISCRETIZATION OF THE NAVIER-STOKES EQUATIONS .....	17
2.1. The convection-diffusion equation .....	17
2.2. Discretization of the convection-diffusion equation .....	18
2.3. Numerical schemes .....	20
2.3.1. Low order numerical schemes .....	21
2.3.2. High order numerical schemes .....	23
2.4. Proposed problems: Convection-Diffusion Equation .....	26
2.4.1. A Two-Dimensional Transient Conduction Problem .....	26
2.4.2. The Smith-Hutton Problem .....	41
2.5. Introduction to the Fractional Step Method .....	52
2.5.1. Time-integration method .....	53
2.6. Proposed problems: Fractional Step Method .....	55
2.6.1. The Driven Cavity .....	55
2.6.2. Differentially Heated Cavity .....	69
2.6.3. Flow around a square cylinder .....	78
3. INTRODUCTION TO TURBULENCE .....	95
3.1. Burgers equation in Fourier space .....	95
3.2. Resolution of Burgers equation .....	97
4. CONCLUSIONS .....	99
4.1. Improvements and future work .....	99
5. SOFTWARE .....	101
5.1. Licenses .....	101

6.	TASKS PLANNIFICATION.....	103
6.1.	Tasks .....	103
6.2.	Dependencies among tasks.....	103
6.3.	Gantt chart .....	103
6.4.	Future tasks .....	103
7.	ENVIRONMENTAL IMPACT.....	105
8.	REFERENCES .....	107

## List of figures

Figure 1: Rectangular finite volume .....	18
Figure 2: Sketch of original variables profile (extracted from [1]) .....	24
Figure 3: Fluxes on a control volume.....	24
Figure 4: Sketch of normalized variables profile (extracted from [1]) .....	26
Figure 5: A Two-Dimensional Transient Conduction Problem. Schema of the proposed problem (by CTTC) .....	27
Figure 6: A Two-Dimensional Transient Conduction Problem. Scheme of the used mesh .....	28
Figure 7: A Two-Dimensional Transient Conduction Problem. Evolution of temperature with time for a 165x120 mesh, time step 1 s and $\beta = 0.5$ .....	36
Figure 8: A Two-Dimensional Transient Conduction Problem. Reference solution for $t = 5000$ s....	36
Figure 9: A Two-Dimensional Transient Conduction Problem. Instantaneous isotherms at $t = 1000$ s for a 165x120 mesh, time step 1 s and $\beta = 0.5$ (temperature in °C).....	37
Figure 10: A Two-Dimensional Transient Conduction Problem. Temperature distribution at $t = 1000$ s for a 165x120 mesh, time step 1 s and $\beta = 0.5$ (temperature in °C).....	38
Figure 11: A Two-Dimensional Transient Conduction Problem. Instantaneous isotherms at $t = 5000$ s for a 165x120 mesh, time step 1 s and $\beta = 0.5$ (temperature in °C).....	38
Figure 12: A Two-Dimensional Transient Conduction Problem. Temperature distribution at $t = 5000$ s for a 165x120 mesh, time step 1 s and $\beta = 0.5$ (temperature in °C).....	39
Figure 13: A Two-Dimensional Transient Conduction Problem. Instantaneous isotherms at $t = 10000$ s for a 165x120 mesh, time step 1 s and $\beta = 0.5$ (temperature in °C) .....	39
Figure 14: A Two-Dimensional Transient Conduction Problem. Temperature distribution at $t = 10000$ s for a 165x120 mesh, time step 1 s and $\beta = 0.5$ (temperature in °C) .....	40
Figure 15: The Smith-Hutton Problem. Schema of the proposed problem (by CTTC).....	41
Figure 16: The Smith-Hutton Problem. Value of $\phi$ at the inlet (equation (68)).....	42
Figure 17: The Smith-Hutton Problem. Isolines of the prescribed velocity field.....	42
Figure 18: The Smith-Hutton Problem. Comparison of results between different mesh sizes using UDS scheme and $\rho/\Gamma = 10$ (1).....	47
Figure 19: The Smith-Hutton Problem. Comparison of results between different mesh sizes using UDS scheme and $\rho/\Gamma = 10$ (2).....	47
Figure 20: The Smith-Hutton Problem. Distribution of $\phi$ in the domain using a 500x250 mesh, UDS scheme and $\rho/\Gamma = 10$ .....	48
Figure 21: The Smith-Hutton Problem. Comparison of results between different mesh sizes using UDS scheme and $\rho/\Gamma = 103$ .....	48
Figure 22: The Smith-Hutton Problem. Distribution of $\phi$ in the domain using a 800x400 mesh, UDS scheme and $\rho/\Gamma = 103$ .....	49
Figure 23: The Smith-Hutton Problem. Comparison of results between different mesh sizes using UDS scheme and $\rho/\Gamma = 106$ .....	49
Figure 24: The Smith-Hutton Problem. Distribution of $\phi$ in the domain using a 800x400 mesh, UDS scheme and $\rho/\Gamma = 106$ .....	50

Figure 25: The Smith-Hutton Problem. Comparison between different numerical schemes $\rho/\Gamma = 106$ .....	50
Figure 26: The Smith-Hutton Problem. Comparison between different numerical schemes $\rho/\Gamma = 103$ .....	51
Figure 27: Convective + Viscous term vector field unique decomposition (extracted from [3]) .....	53
Figure 28: The Driven Cavity. Scheme of the problem (extracted from [5]) .....	55
Figure 29: The Driven Cavity. U-component nodes (blue) superposed to the pressure mesh.....	57
Figure 30: The Driven Cavity. V-component nodes (red) superposed to the pressure mesh.....	57
Figure 31: The Driven Cavity. Schema of the displaced control volume .....	58
Figure 32: The Driven Cavity. Obtained results for U velocity component in the vertical mid-plane with different mesh sizes and $Re = 100$ .....	64
Figure 33: The Driven Cavity. Obtained results for V velocity component in the horizontal mid-plane with different mesh sizes and $Re = 100$ .....	64
Figure 34: The Driven Cavity. Obtained results for U velocity component in the vertical mid-plane with different mesh sizes and $Re = 5000$ .....	65
Figure 35: The Driven Cavity. Obtained results for V velocity component in the horizontal mid-plane with different mesh sizes and $Re = 5000$ .....	65
Figure 36: The Driven Cavity. Obtained results for U velocity component in the vertical mid-plane with different mesh sizes and $Re = 10000$ .....	66
Figure 37: The Driven Cavity. Obtained results for V velocity component in the horizontal mid-plane with different mesh sizes and $Re = 10000$ .....	66
Figure 38: The Driven Cavity. Velocity module and streamlines for different Reynolds numbers ....	68
Figure 39: Differentially Heated Cavity. Scheme of the problem (extracted from [8]) .....	69
Figure 40: Differentially Heated Cavity. Isolines of the temperature field with a 125x125 mesh .....	74
Figure 41: Differentially Heated Cavity. Isolines of the U component field with a 125x125 mesh .....	74
Figure 42: Differentially Heated Cavity. Isolines of the V component field with a 125x125 mesh .....	75
Figure 43: Differentially Heated Cavity. Velocity field and streamlines with a 125x125 mesh .....	76
Figure 44: Differentially Heated Cavity. Isolines of the vorticity field with a 125x125 mesh .....	76
Figure 45: Differentially Heated Cavity. Nusselt number distribution at the hot wall for different Rayleigh numbers with a 125x125 mesh.....	77
Figure 46: Differentially Heated Cavity. U velocity component in the vertical mid-plane for different Rayleigh numbers with a 125x125 mesh.....	77
Figure 47: Differentially Heated Cavity. V velocity component in the horizontal mid-plane for different Rayleigh numbers with a 125x125 mesh (1) .....	77
Figure 48: Differentially Heated Cavity. V velocity component in the horizontal mid-plane for different Rayleigh numbers with a 125x125 mesh (2) .....	77
Figure 49: Square cylinder. Scheme of the problem (case 1) (extracted from [10]) .....	78
Figure 50: Square cylinder. Scheme of the problem (case 2) (extracted from [11]) .....	79
Figure 51: Square cylinder. Representation of a control volume touching the obstacle with its east face .....	81
Figure 52: Square cylinder. Aerodynamic forces on an elemental area (extracted from [12]).....	81

Figure 53: Square cylinder. Non-equidistant mesh used in the case 2 (close view near the obstacles) .....	83
Figure 54: Square cylinder (case 1). Evolution of the steady drag coefficient with the Reynolds number.....	84
Figure 55: Square cylinder (case 1). Evolution of the recirculation length with the Reynolds number .....	84
Figure 56: Square cylinder (case 1). Evolution of the time-averaged drag coefficient with the Reynolds number .....	84
Figure 57: Square cylinder (case 1). Evolution of the difference between maximum and minimum drag coefficient with the Reynolds number .....	84
Figure 58: Square cylinder (case 1). Evolution of the difference between maximum and minimum lift coefficient with the Reynolds number .....	84
Figure 59: Square cylinder (case 1). Evolution of the Strouhal number with the Reynolds number	84
Figure 60: Square cylinder (case 1). Streamlines around the obstacle for different Reynolds numbers.....	85
Figure 61: Square cylinder (case 1). Isolines of the velocity field for different Reynolds numbers ..	86
Figure 62: Square cylinder (case 1). Pseudo-pressure field for different Reynolds numbers.....	87
Figure 63: Square cylinder (case 2). Evolution of the time-averaged drag coefficient with the separation ratio s/d .....	88
Figure 64: Square cylinder (case 2). Evolution of the root mean square value of the lift coefficient with the separation ratio s/d .....	88
Figure 65: Square cylinder (case 2). Evolution of the Strouhal number with the separation ratio s/d .....	88
Figure 66: Square cylinder (case 2). Evolution of instantaneous streamlines around the two obstacles with the separation ratio s/d .....	89
Figure 67: Square cylinder (case 2). Evolution of instantaneous velocity field around the two obstacles with the separation ratio s/d .....	91
Figure 68: The Burguers equation. Reference solution for the energy spectrum of the steady-state solution of the Burguers equation with $Re = 40$ (extracted from [16]) .....	97
Figure 69: The Burguers equation. Results obtained with the simulation of the Burguers equation for $Re = 40$ .....	98



## List of tables

Table 1: Parameters to replace in the convection-diffusion equation in order to reproduce the governing equations.....	17
Table 2: Value of $A( P )$ for different low numerical schemes .....	23
Table 3: A Two-Dimensional Transient Conduction Problem. Problem coordinates.....	27
Table 4: A Two-Dimensional Transient Conduction Problem. Physical properties.....	27
Table 5: A Two-Dimensional Transient Conduction Problem. Boundary conditions .....	27
Table 6: A Two-Dimensional Transient Conduction Problem. Discretization coefficients of boundary nodes .....	30
Table 7: A Two-Dimensional Transient Conduction Problem. Discretization coefficients of corner nodes .....	30
Table 8: A Two-Dimensional Transient Conduction Problem. Discretization coefficients of inner nodes .....	32
Table 9: A Two-Dimensional Transient Conduction Problem. Some results for a 165x120 mesh and $\beta = 0.5$ .....	35
Table 10: The Smith-Hutton Problem. Cases to be solved .....	42
Table 11: The Smith-Hutton Problem. Discretization coefficients of boundary nodes .....	43
Table 12: The Smith-Hutton Problem. Discretization coefficients of corner nodes .....	43
Table 13: The Smith-Hutton Problem. Reference solution .....	46
Table 14: The Driven Cavity. Discretization coefficients of boundary nodes of the pressure field ...	59
Table 15: The Driven Cavity. Discretization coefficients of inner nodes of the pressure field .....	60
Table 16: The Driven Cavity. Reference solution (U-component) .....	63
Table 17: The Driven Cavity. Reference solution (V-component) .....	63
Table 18: Differentially Heated Cavity. Discretization coefficients of boundary nodes of the temperature field .....	72
Table 19: Differentially Heated Cavity. Reference results of the problem (see [8]) .....	73
Table 20: Differentially Heated Cavity. More accurate obtained results (125x125 meh) .....	73
Table 21: Square cylinder. Discretization coefficients of boundary nodes of the pressure field .....	80
Table 22: Square cylinder (case 2). Mesh sizes according to s/d ratio .....	87
Table 23: Dependencies among tasks .....	103
Table 24: Initial Gantt chart .....	103
Table 25: Final Gantt chart.....	103



## 1. INTRODUCTION

### 1.1. Aim of the study

The main goal of this study is the development of a code based on CFD (Computational Fluid Dynamics) and HT (Heat Transfer), its validation and its verification using some solved problems as a reference and using all the information and documents provided by the *Centre Tecnològic de Transferència de Calor* (CTTC) from the Universitat Politècnica de Catalunya. Furthermore, the physics behind the problems and the development of the code must be understood and, therefore, a theory study is done with all the information provided by CTTC.

### 1.2. Scope of the study

- Development of the code
  - i) Study and discretization of the Navier – Stokes equations
  - ii) Study of some resolution methods
  - iii) Study of turbulence (if applicable)
  - iv) Application of the code to some proposed problems. Validation and verification of the code
- Application to an engineering problem
  - i) Selection of an adequate engineering problem to solve with the developed code
  - ii) Identification and measurement of the convenience of solving the Navier – Stokes equations in the field of the selected engineering problem
- Results
  - i) Evaluation and conclusions of the results obtained in the simulation of the selected engineering problem with the developed code
  - ii) Development of a planning for future improvements of the code in order to achieve better results with the selected engineering problem or others (if applicable)

### 1.3. Requirements

The code must be developed by oneself. All the information used in order to develop the code must be referenced according to the intellectual property laws.

### 1.4. Justification

Engineering fields such as structures or fluid dynamics are using computational methods on many of their projects. The computers are becoming more and more powerful and, therefore, the importance of these methods is increasing, since more complicated cases can be solved with relative short time.

In the middle of the 20th century, the computational methods were not normally used because of the low computational power of the computers of this epoch and the simulations were basically done with experimental testing. Nowadays, these methods are one of the most important branches of the engineering simulation. If the code is well implemented, the obtained results are very close to the reality and they serve as support for further experimental testing.

The aerospace engineering field needs of computational methods in almost every one of its branches, as aerodynamics need of the resolution of the Navier – Stokes equations to compute the velocity field around the wings or the fuselage and also the energy transfer (e.g. temperature) between the air and the airplane. With this, it is possible to obtain really accurate estimations of the aerodynamic parameters and the flight conditions of the designed airplane or even the necessary thermal isolation of a spacecraft when entering the atmosphere of a planet. Therefore, this study will serve as an important introduction to the resolution of these equations that are constantly needed in the world of aerospace engineering.

## 2. STUDY AND DISCRETIZATION OF THE NAVIER-STOKES EQUATIONS

### 2.1. The convection-diffusion equation

The resolution of the problem of heat transfer by convection is obtained solving a state equation (relation between pressure, temperature and density) coupled with conservative equations of mass, linear momentum and energy: they are the governing equations (found in [1]). Also constitutive relations are required, such as Stokes' law or Fourier's law.

The unknowns of this system of equations are the temperature, the pressure and the components of the velocity field. In order to close the problem, boundary and initial conditions are required too.

Then, the two strong couplings that characterize this system of equations are:

- Pressure-velocity (the pressure is the field that makes the velocity accomplish the mass conservation equation for incompressible flows).
- Temperature-velocity (only present for natural convection, mixed convection or temperature dependence in physical properties).

All these governing equations can be summarized in the convection-diffusion equation:

$$\frac{\partial \rho \phi}{\partial t} + \nabla \cdot (\rho \vec{v} \phi) = \nabla \cdot (\Gamma \nabla \phi) + S \quad (1)$$

Where  $\phi$  is the property to be evaluated. This equation states that the accumulation of  $\phi$  plus the net convective flow has to be equal to the net diffusive flow plus the generation of  $\phi$  per unit of volume.

Now, it is possible to express the governing equations using the convection-diffusion equation. The following table (see [1]) shows the variables that must be taken to express each governing equation.

Equation	$\phi$	$\Gamma$	S
Continuity	1	0	0
Momentum in x direction	u	$\mu$	$-\partial p_d / \partial x$
Momentum in y direction	v	$\mu$	$-\partial p_d / \partial y + \rho g \beta (T - T_\infty)$
Energy (constant $c_p$ )	T	$\lambda / c_p$	$\Phi / c_p$

Table 1: Parameters to replace in the convection-diffusion equation in order to reproduce the governing equations

The convection-diffusion equation can be simplified assuming constant physical properties ( $\rho$  and  $\Gamma$ ).

Then, equation (1) can be rewritten as follows:

$$\frac{\partial \phi}{\partial t} + \nabla \cdot (\vec{v}\phi) = \frac{\Gamma}{\rho} \Delta \phi + \frac{S}{\rho} \quad (2)$$

If it is an incompressible flow, another simplification can be done:

$$\nabla \cdot (\vec{v}\phi) = (\vec{v} \cdot \nabla)\phi + \phi \nabla \cdot \vec{v} \rightarrow \nabla \cdot \vec{v} = 0 \quad (3)$$

And equation (2) can be rewritten as:

$$\frac{\partial \phi}{\partial t} + (\vec{v} \cdot \nabla)\phi = \frac{\Gamma}{\rho} \Delta \phi + \frac{S}{\rho} \quad (4)$$

## 2.2. Discretization of the convection-diffusion equation

The selected discretization method for solving the convection-diffusion equation is the Finite Volume Method (FVM). It presents more advantages when solving the equation than other discretization methods like Finite Difference Method (FDM) or Finite Element Method (FEM, widely used in structures).

In order to discretize the convection-diffusion equation using the FVM, the equation has to be integrated into a polygonal finite volume (rectangular, triangular, etc.). This study is focused on rectangular meshes and, therefore, the used finite volume follows the scheme showed in *Figure 1*.

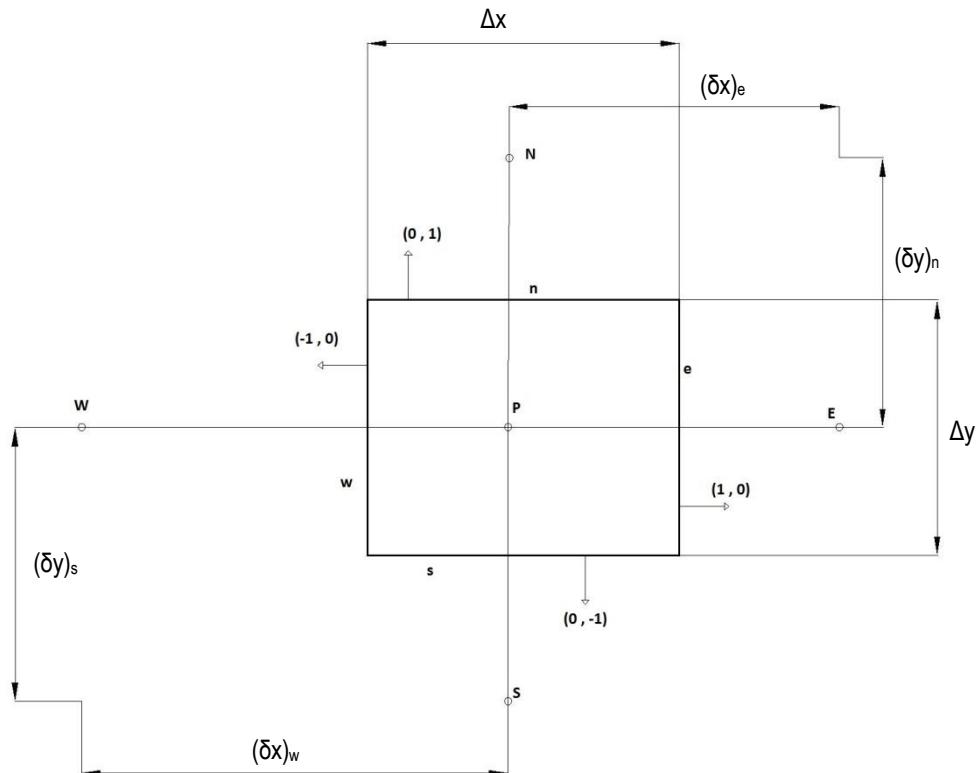


Figure 1: Rectangular finite volume

The integrals are done using the Gauss Theorem shown in equation (5), which relates the divergence of a vector with the scalar product between this vector and the normal vector of the boundary of an arbitrary domain.

$$\int_{\Omega} \nabla \cdot \vec{a} \, d\Omega = \int_{\partial\Omega} \vec{a} \cdot \vec{n} \, dS \quad (5)$$

Applying this theorem to the convection-diffusion equation (1), it yields to:

$$\int_{\Omega} \left( \frac{\partial \rho \phi}{\partial t} + \nabla \cdot (\rho \vec{v} \phi) = \nabla \cdot (\Gamma \nabla \phi) + S \right) d\Omega \quad (6)$$

Then, the different terms are integrated using the following hypothesis [1]:

- The convective and diffusive flows are considered to be constant through each face of the control volume.
- $(\text{spatial deviation})^n = (\text{spatial deviation})^{n+1}$
- $(\text{spatial deviation})_w = (\text{spatial deviation})_e$
- $(\text{spatial deviation})_s = (\text{spatial deviation})_n$
- The source term is supposed constant along the control volume with the value at  $S_p^{n+1}$ .
- A fully implicit scheme is used\*.

\*The first validation problem (section 2.4.1) uses a parameter to choose the temporal scheme.

The temporal term, or accumulation of  $\phi$ , can be discretized as follows:

$$\int_{\Omega} \frac{\partial \rho \phi}{\partial t} \, d\Omega \cong \int_{\Omega} \frac{(\rho \phi)_p^{n+1} - (\rho \phi)_p^n}{\Delta t} \, d\Omega \cong \frac{(\rho \phi)_p^{n+1} - (\rho \phi)_p^n}{\Delta t} \Delta x \Delta y \quad (7)$$

Then, in order to discretize the convective term, the Gauss Theorem is applied:

$$\begin{aligned} \int_{\Omega} \nabla \cdot (\rho \vec{v} \phi) \, d\Omega &= \int_{\partial\Omega} (\rho \vec{v} \phi) \cdot \vec{n} \, dS \cong \\ &\cong [(\rho u \phi)_e^{n+1} - (\rho u \phi)_w^{n+1}] \cdot \Delta y + [(\rho v \phi)_n^{n+1} - (\rho v \phi)_s^{n+1}] \cdot \Delta x \end{aligned} \quad (8)$$

Also, for the discretization of the diffusive term, the Gauss Theorem is used:

$$\int_{\Omega} \nabla \cdot (\Gamma \nabla \phi) \, d\Omega = \int_{\partial\Omega} (\Gamma \nabla \phi) \cdot \vec{n} \, dS \cong$$

$$\cong \left[ \left( \Gamma \frac{\partial \phi}{\partial x} \right)_e^{n+1} - \left( \Gamma \frac{\partial \phi}{\partial x} \right)_w^{n+1} \right] \cdot \Delta y + \left[ \left( \Gamma \frac{\partial \phi}{\partial y} \right)_n^{n+1} - \left( \Gamma \frac{\partial \phi}{\partial y} \right)_s^{n+1} \right] \cdot \Delta x \quad (9)$$

The last term to discretize is the source term and, as it has been said before, it is considered constant inside the control volume:

$$\int_{\Omega} S \, d\Omega \cong \int_{\Omega} S_P^{n+1} \, d\Omega \cong S_P^{n+1} \Delta x \Delta y \quad (10)$$

The discretized convection-diffusion equation is obtained relating the equations (7) to (10):

$$\begin{aligned} & \frac{(\rho \phi)_P^{n+1} - (\rho \phi)_P^n}{\Delta t} \Delta x \Delta y + [(\rho u \phi)_e^{n+1} - (\rho u \phi)_w^{n+1}] \cdot \Delta y + \\ & + [(\rho v \phi)_n^{n+1} - (\rho v \phi)_s^{n+1}] \cdot \Delta x = \left[ \left( \Gamma \frac{\partial \phi}{\partial x} \right)_e^{n+1} - \left( \Gamma \frac{\partial \phi}{\partial x} \right)_w^{n+1} \right] \cdot \Delta y + \\ & + \left[ \left( \Gamma \frac{\partial \phi}{\partial y} \right)_n^{n+1} - \left( \Gamma \frac{\partial \phi}{\partial y} \right)_s^{n+1} \right] \cdot \Delta x + S_P^{n+1} \Delta x \Delta y \end{aligned} \quad (11)$$

### 2.3. Numerical schemes

Numerical schemes are used to compute the value of the dependent variable  $\phi$  at the cell faces. This problem arises because the discretized convection-diffusion equation (11) needs of the evaluation of the convective and diffusive terms at the cell faces, while the dependent variable is known at the center of the control volume.

According to its order (number of neighboring nodes used to evaluate the variable  $\phi$  at the boundary of the control volume), numerical schemes are classified in low order or high order schemes. When computing the value of the variable at the cell face, it is assumed that the cell face is in the middle between two nodal points (same distance between cell face and nodes). Otherwise, it would be necessary to introduce geometric variables to take into account the different distance between cell faces and nodes.

In order to compute the conductive flux at the cell faces, it can be calculated as an arithmetic mean using the values of the dependent variable in the cell center and the distance between nodes.

$$\left( \frac{\partial \phi}{\partial y} \right)_n = \frac{\phi_N - \phi_P}{\delta y_n} \quad (12)$$

$$\left( \frac{\partial \phi}{\partial y} \right)_s = \frac{\phi_P - \phi_S}{\delta y_s} \quad (13)$$

$$\left( \frac{\partial \phi}{\partial x} \right)_e = \frac{\phi_E - \phi_P}{\delta x_e} \quad (14)$$

$$\left( \frac{\partial \phi}{\partial x} \right)_w = \frac{\phi_P - \phi_W}{\delta x_w} \quad (15)$$

### 2.3.1. Low order numerical schemes

As written before, low order numerical schemes approximate the value of the dependent variable at the cell faces using the nearest neighboring nodes of the studied control volume (see *Figure 1*). These approximations can be first or second order approximations, depending on the used scheme.

The most significant low order numerical schemes are the following (see [1] and [2]):

- **Upwind Difference Scheme (UDS):** This numerical scheme is a first order scheme. In order to evaluate the value of  $\phi$  at the cell face, this scheme evaluates the direction of the convective flow. The value of the dependent variable at the cell face is equal to the value of the variable at the center of the control volume from which the flow comes.

This scheme can be summarized with the following equations.

$$\phi_n = \begin{cases} \phi_N & \text{if } F_n < 0 \\ \phi_P & \text{if } F_n > 0 \end{cases} \quad (16)$$

$$\phi_s = \begin{cases} \phi_S & \text{if } F_s > 0 \\ \phi_P & \text{if } F_s < 0 \end{cases} \quad (17)$$

$$\phi_e = \begin{cases} \phi_E & \text{if } F_e < 0 \\ \phi_P & \text{if } F_e > 0 \end{cases} \quad (18)$$

$$\phi_w = \begin{cases} \phi_W & \text{if } F_w > 0 \\ \phi_P & \text{if } F_w < 0 \end{cases} \quad (19)$$

Although this scheme has been widely used due to its simplicity, it has a disadvantage: it moves the values of the dependent variable a half of a control volume and, therefore, it acts as a false transporter of the properties.

- **Central Difference Scheme (CDS):** This numerical scheme is a second order scheme. It uses an arithmetic mean between the two values of the dependent variable calculated in the center of the respective control volumes when evaluating the value of the variable at the cell face. This scheme can be summarized as follows:

$$\phi_n = \frac{1}{2}(\phi_P + \phi_N) \quad (20)$$

$$\phi_s = \frac{1}{2}(\phi_P + \phi_S) \quad (21)$$

$$\phi_e = \frac{1}{2}(\phi_P + \phi_E) \quad (22)$$

$$\phi_w = \frac{1}{2}(\phi_P + \phi_W) \quad (23)$$

- **Hybrid Difference Scheme (HDS):** This scheme is a combination of CDS and UDS schemes. It uses CDS for low velocities and UDS for high velocities, controlling it with the Peclet number. For Peclet numbers above 2, the CDS becomes unstable and usually a stable solution cannot be reached (but when reached, the obtained solution has no sense). Therefore, for Peclet numbers above 2, UDS scheme is used.
- **Exponential Difference Scheme (EDS):** This scheme is also a second order scheme but more accurate than CDS or HDS schemes. The evaluation of the variable at the cell face comes from the exact solution of the convection-diffusion equation in one-dimensional case, null source term and steady problem. As this scheme is more accurate, it also takes more computational efforts and time.
- **Power Law Difference Scheme (PLDS):** This scheme is a second order scheme that evaluates the variable at the cell face using an approximation of EDS scheme by a polynomial of fifth degree. It is simpler than EDS, but, even so, it takes more computational efforts than UDS, CDS or HDS schemes.

Introducing these numerical schemes in the discretized convection-diffusion equation (equation (11)), an algebraic equation for each control volume is obtained with the following form:

$$a_P \phi_P^{n+1} = a_N \phi_N^{n+1} + a_S \phi_S^{n+1} + a_E \phi_E^{n+1} + a_W \phi_W^{n+1} + b_P \quad (24)$$

The coefficients of this equation can be evaluated as follows (see [1]):

$$a_N = D_n \cdot A(|P_n|) + \max(-F_n, 0) \quad (25)$$

$$a_S = D_s \cdot A(|P_s|) + \max(F_s, 0) \quad (26)$$

$$a_E = D_e \cdot A(|P_e|) + \max(-F_e, 0) \quad (27)$$

$$a_W = D_w \cdot A(|P_w|) + \max(F_w, 0) \quad (28)$$

$$a_P = a_N + a_S + a_E + a_W + \rho_P^n \cdot \frac{\Delta x \Delta y}{\Delta t} \quad (29)$$

$$b_P = \rho_P^n \cdot \frac{\Delta x \Delta y}{\Delta t} \cdot \phi_P^n + S_P^{n+1} \Delta x \Delta y \quad (30)$$

Where:

$$D_n = \frac{\Gamma_n \Delta x}{(\delta y)_n}; D_s = \frac{\Gamma_s \Delta x}{(\delta y)_s}; D_e = \frac{\Gamma_e \Delta y}{(\delta x)_e}; D_w = \frac{\Gamma_w \Delta y}{(\delta x)_w} \quad (31)$$

$$F_n = (\rho v)_n \Delta x; F_s = (\rho v)_s \Delta x; F_e = (\rho u)_e \Delta y; F_w = (\rho u)_w \Delta y \quad (32)$$

The Peclet number evaluated at the face of the control volume (f) is:

$$P_f = \frac{F_f}{D_f} \quad (33)$$

With this generic way to obtain the coefficients, the value of  $A(|P|)$  is computed according to the selected numerical scheme as shown in *Table 2* (see [1]).

Numerical Scheme	$A( P )$
UDS	1
CDS	$1 - 0.5( P )$
HDS	$\max(0, (1 - 0.5( P )))$
EDS	$ P /(e^{ P } - 1)$
PLDS	$\max(0, (1 - 0.5( P ))^5)$

Table 2: Value of  $A(|P|)$  for different low numerical schemes

### 2.3.2. High order numerical schemes

It is important to develop new numerical schemes that improve the accuracy of the low order numerical schemes in order to avoid the problems derived from the usage of first and second order schemes (numerical diffusion errors, etc.). These improved numerical schemes can be obtained using more than two nodal values when computing the dependent variable at the cell face. They are called high order numerical schemes and can be introduced into the general formulation seen in the section 2.3.1, adding an extra term called deferred term ( $b_{de}$ ) that acts as if it was a source term. The reformulation computes the coefficients of equation (24) using the UDS scheme and this equation can be rewritten as follows:

$$a_P \phi_P^{n+1} = a_N \phi_N^{n+1} + a_S \phi_S^{n+1} + a_E \phi_E^{n+1} + a_W \phi_W^{n+1} + b_P + b_{de} \quad (34)$$

Then, the deferred term is computed as seen in the equation (35), where  $F_f$  represents the convective flow in the cell face  $f$ ,  $\phi_f^{UDS}$  represents the variable evaluated at the cell face  $f$  using UDS scheme and  $\phi_f^{HS}$  represents the variable evaluated at the cell face  $f$ , but using a high order numerical scheme.

$$b_{de} = F_n(\phi_n^{UDS} - \phi_n^{HS}) - F_s(\phi_s^{UDS} - \phi_s^{HS}) + F_e(\phi_e^{UDS} - \phi_e^{HS}) - F_w(\phi_w^{UDS} - \phi_w^{HS}) \quad (35)$$

In order to evaluate the variable at the cell face using a high order numerical scheme independently of the flux direction, some new variables must be introduced (see *Figure 2*). It can be seen how the high order numerical schemes use more nodes when computing the value of the variable at the cell face:

- $\phi_C$  is the value of the variable at the nearest grid point of the evaluated face, in the upwind direction of the flow.
- $\phi_U$  is the value of the variable at the grid point above  $\phi_C$ .
- $\phi_D$  is the value of the variable at the nearest grid point of the evaluated face, on the downstream direction of the flow.

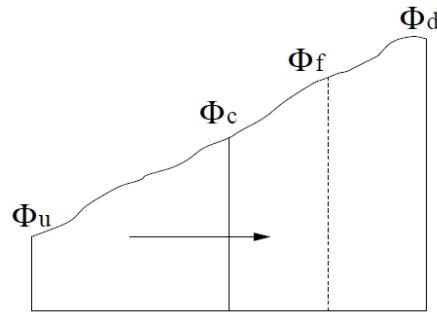


Figure 2: Sketch of original variables profile (extracted from [1])

The most used numerical schemes are the following (see [1]):

- **Upwind Difference Scheme (UDS) of 2<sup>nd</sup> order:** This scheme is an improvement of the UDS of first order. It uses again the value of the variable in the central node of the control volume from where the flow comes, but it also considers the value of the variable in the previous node of the flow direction. The scheme consists of a lineal extrapolation between  $\phi_c$  and  $\phi_u$ . The computation of the value of the variable at the cell face can be summarized with the equation (36).

$$\phi_f = \frac{1}{2}(3\phi_c - \phi_u) \quad (36)$$

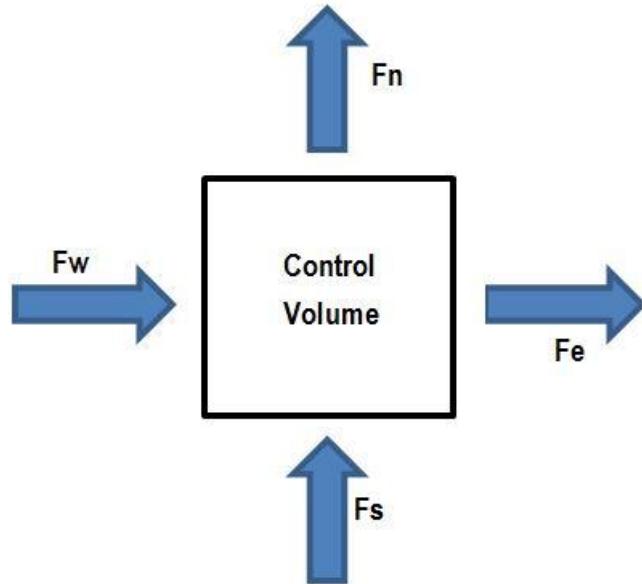


Figure 3: Fluxes on a control volume

If the fluxes follow the direction shown in *Figure 3*, then, the variable evaluated at the cell faces would be as follows:

$$\phi_n = \frac{1}{2}(3\phi_P - \phi_S) \quad (37)$$

$$\phi_s = \frac{1}{2}(3\phi_S - \phi_{SS}) \quad (38)$$

$$\phi_e = \frac{1}{2}(3\phi_P - \phi_W) \quad (39)$$

$$\phi_w = \frac{1}{2}(3\phi_W - \phi_{WW}) \quad (40)$$

- **QUICK:** This high order numerical scheme computes the value of the variable at the cell face using a quadratic interpolation (parabola) between  $\phi_C$ ,  $\phi_U$  and  $\phi_D$ . It is a third order numerical scheme. This type of numerical scheme complicates the system of equations to be solved and, therefore, more computational efforts and time are needed, but it improves the accuracy and the results. The variable at the cell face can be evaluated with the equation (41).

$$\phi_f = \frac{1}{8}(6\phi_C + 3\phi_D - \phi_U) \quad (41)$$

If the fluxes follow the directions shown in *Figure 3*, then the values of the variable at the cell faces would be computed as follows:

$$\phi_n = \frac{1}{8}(6\phi_P + 3\phi_N - \phi_S) \quad (42)$$

$$\phi_s = \frac{1}{8}(6\phi_S + 3\phi_P - \phi_{SS}) \quad (43)$$

$$\phi_e = \frac{1}{8}(6\phi_P + 3\phi_E - \phi_W) \quad (44)$$

$$\phi_w = \frac{1}{8}(6\phi_W + 3\phi_P - \phi_{WW}) \quad (45)$$

Usually, the high order numerical schemes are implemented using normalized variables. This solution arises when bounding the numerical schemes. One of the main problems of high order numerical schemes (or accurate ones) is the instability. If these numerical schemes are bounded ( $\phi_f$  is set between nearest nodal grid points values), the problem of instability can be solved.

The dependent variable can be normalized as shown in equation (46). If this value is normalized,  $\overline{\phi}_f$  depends on a function of  $\overline{\phi}_C$  and, if it is bounded, it can only take a value between 0 and 1, where 0 corresponds to the value of  $\overline{\phi}_U$  and 1 corresponds to the value of  $\overline{\phi}_D$  (see *Figure 4*).

$$\overline{\phi}_f = \frac{\phi_f - \phi_U}{\phi_D - \phi_U} \quad (46)$$

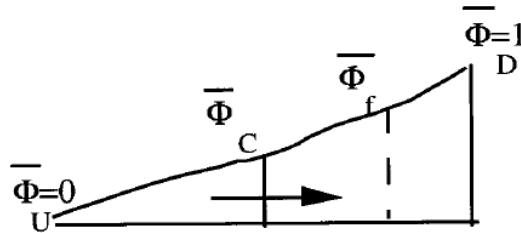


Figure 4: Sketch of normalized variables profile (extracted from [1])

Second and third order numerical schemes are not always bounded (the normalized value of the variable at the cell face is not always between 0 and 1) and they will have instability problems depending on the situation. In order to avoid these problems, there have been developed some schemes that use different evaluations of the variable at the cell face according to the value of  $\bar{\phi}_c$ .

For example, the **SMART** scheme is a combination of two to fourth order numerical schemes and computes the value of the variable as follows:

$$\bar{\phi}_f = 3\bar{\phi}_c \quad \text{if } 0 < \bar{\phi}_c < \frac{1}{6} \quad (47a)$$

$$\bar{\phi}_f = \frac{3}{8} + \frac{3}{4}\bar{\phi}_c \quad \text{if } \frac{1}{6} < \bar{\phi}_c < \frac{5}{6} \quad (47b)$$

$$\bar{\phi}_f = 1 \quad \text{if } \frac{5}{6} < \bar{\phi}_c < 1 \quad (47c)$$

$$\bar{\phi}_f = \bar{\phi}_c \quad \text{otherwise} \quad (47d)$$

## 2.4. Proposed problems: Convection-Diffusion Equation

### 2.4.1. A Two-Dimensional Transient Conduction Problem

#### 2.4.1.1. Objective

This problem is proposed by CTTC as an example of the convection-diffusion equation. The objective of the problem is to solve the heat transfer by conduction of a bi-dimensional body.

#### 2.4.1.2. Problem definition

The problem consists in a very long rod and, therefore, the hypothesis of 2D problem can be used. It is made of four different materials with different physical properties and the boundary conditions are different according to the side of the rod.

This problem corresponds to a transient heat transfer situation because of the changing right-side boundary condition according to time (see *Table 5*). In particular, it does not have convection (solid materials) and all the heat transfer occurs by diffusion.

Furthermore, the temperature evolution during 10000 seconds must be computed in two different points of the domain (at location (0.65, 0.56) and (0.74, 0.72)). The initial temperature ( $t = 0$  s) in all the domain is 8.00 °C.

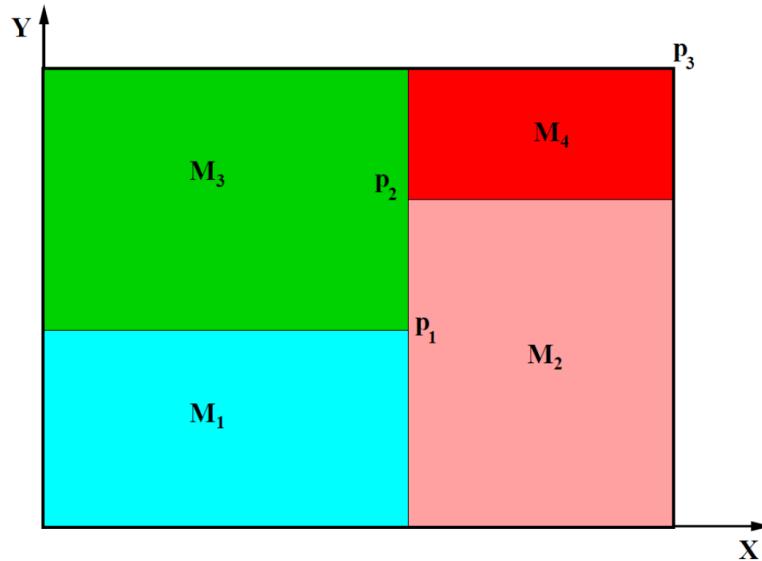


Figure 5: A Two-Dimensional Transient Conduction Problem. Schema of the proposed problem (by CTTC)

Points	x [m]	y [m]
p <sub>1</sub>	0.50	0.40
p <sub>2</sub>	0.50	0.70
p <sub>3</sub>	1.10	0.80

Table 3: A Two-Dimensional Transient Conduction Problem. Problem coordinates

Material	$\rho$ [ $kg/m^3$ ]	$c_p$ [ $J/kgK$ ]	$\lambda$ [ $W/mK$ ]
M1	1500.00	750.00	170.00
M2	1600.00	770.00	140.00
M3	1900.00	810.00	200.00
M4	2500.00	930.00	140.00

Table 4: A Two-Dimensional Transient Conduction Problem. Physical properties

Cavity wall	Boundary conditions
Bottom	Isotherm at $T_{base} = 23.00^\circ\text{C}$
Top	Uniform $Q_{flow} = 60.00 \text{ W/m}$ length
Left	In contact with a fluid at $T_g = 33.00^\circ\text{C}$ and heat transfer coefficient $9.00 \text{ W/m}^2\text{K}$
Right	Uniform temperature $T = 8.00 + 0.005t$ °C (where t is the time in seconds)

Table 5: A Two-Dimensional Transient Conduction Problem. Boundary conditions

#### 2.4.1.3. Numerical discretization

As explained before, the main objective of the discretization is to achieve a system of equations like equation (24), where all the relations between nodes are expressed using some coefficients. From

now on, the problem will be treated with physical meaning. Then, the variable  $\phi$  is considered to be the temperature of the body.

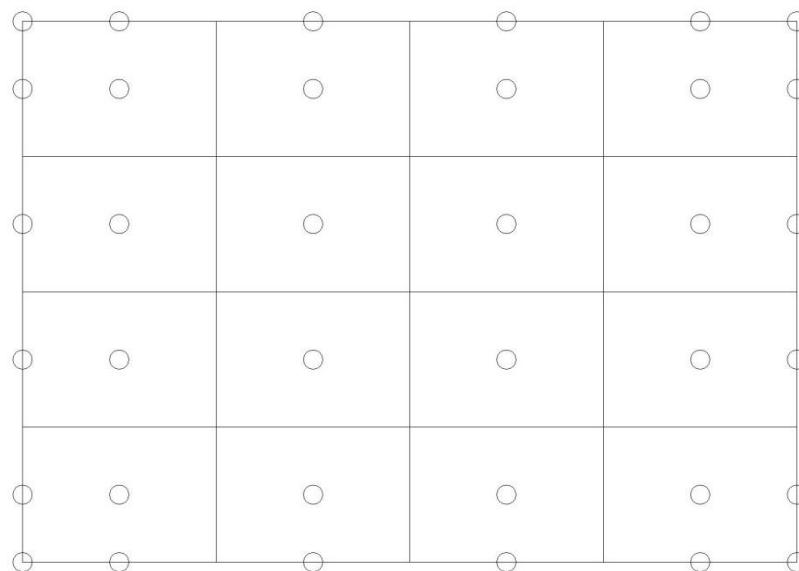
First of all, it is useful to introduce the  $\beta$  coefficient in the discretization, in order to be able to change the temporal discretization of the problem. Then, the discretized convection-diffusion equation, using physical meanings of the variables and coefficients, is written as:

$$\frac{\rho V c_p}{\Delta t} (T_P^{n+1} - T_P^n) = \beta \sum \dot{Q}_P^{n+1} + (1 - \beta) \sum \dot{Q}_P^n \quad (48)$$

Where  $\sum \dot{Q}_P$  are the heat fluxes around the node P. In addition, according to the value of the  $\beta$  coefficient, there are different temporal numerical schemes:

- Explicit ( $\beta = 0$ )
- Implicit ( $\beta = 1$ )
- Crank-Nicolson ( $\beta = 0.5$ )

Now, it is important to think about the proper mesh. For this discretization, an equidistant rectangular mesh can be used. The domain can be divided in equal control volumes with a node in the center of each one and more nodes can be collocated in the boundary of the domain (with no control volume associated) in order to improve the insertion of the boundary conditions into the system of equations (see *Figure 6*).



**Figure 6: A Two-Dimensional Transient Conduction Problem. Scheme of the used mesh**

One of the particularities of the mesh for this problem is that it is better if the mesh fits the space occupied by the materials, meaning that no control volume is shared between two or more materials. Then, the mesh generation code must be developed taking into account this particularity.

Another important particularity of this problem is the changing zone between materials with different properties. These changes only affect the thermal conductivity ( $\lambda$ ) between materials. The density ( $\rho$ ) and the specific heat capacity ( $c_p$ ) are computed into the whole control volumes and according to the designed mesh, no control volumes are shared between materials.

When computing the heat fluxes of a control volume, the heat flux by conduction is described by Fourier's law (see equation (49)); note the similarities with the convection-diffusion equation. The boundary condition regarding the left cavity wall is considered to be as described in equation (50).

$$\dot{q} = -\lambda \cdot \frac{\partial T}{\partial x} \left[ \frac{W}{m} \right] \rightarrow \dot{Q} = -\lambda \cdot \frac{\Delta T}{\Delta x} \cdot \Delta y [W] \quad (49)$$

$$\dot{q} = \alpha \Delta T [W/m] \quad (50)$$

The proposed solution to compute the thermal conductivity coefficient at the cell faces of the control volumes shared by two materials has been the elaboration of a harmonic mean between the two coefficients corresponding to each material.

As an example, if two different nodes (P and E) from two different materials (A and B) share a common face (e, from east) between the two corresponding control volumes, the heat flux entering the face has to be equal to the heat flux leaving the face ( $\dot{q}_e^- = \dot{q}_e^+ = \dot{q}_e$ ). This situation can be written as:

$$\begin{cases} \dot{q}_e = \dot{q}_e^- = -\frac{\lambda_A}{d_{Pe}} \cdot (T_e - T_P) \\ \dot{q}_e = \dot{q}_e^+ = -\frac{\lambda_B}{d_{Ee}} \cdot (T_E - T_e) \end{cases} \rightarrow \dot{q}_e = \frac{d_{PE}}{\frac{d_{Pe}}{\lambda_A} + \frac{d_{Ee}}{\lambda_B}} \cdot \frac{T_E - T_P}{d_{PE}} \quad (51)$$

Then, the thermal conductivity coefficient of the cell face can be identified from equation (51):

$$\lambda_e = \frac{d_{PE}}{\frac{d_{Pe}}{\lambda_A} + \frac{d_{Ee}}{\lambda_B}} \quad (52)$$

The same procedure can be applied to compute the coefficient for the other directions (north, south and west).

At this point, the discretization coefficients of the boundary nodes and the nodes in the corners can be studied. All the information regarding the value assigned to the coefficients is found in *Table 6* and *Table 7*.

Bottom wall	Value
$a_P$	1
$a_E$	0
$a_W$	0
$a_N$	0
$a_S$	0
$b_P$	$T_{base}$
Top wall	Value
$a_P$	$\frac{2 \cdot \beta \cdot \lambda_S \cdot \Delta x}{\Delta y}$
$a_E$	0
$a_W$	0
$a_N$	0
$a_S$	$\frac{2 \cdot \beta \cdot \lambda_S \cdot \Delta x}{\Delta y}$
$b_P$	$\dot{Q}_{flow} \cdot \Delta x + (1 - \beta) \cdot \left( \frac{-2 \cdot \lambda_S \cdot \Delta x}{\Delta y} \right) \cdot T_P^n + (1 - \beta) \cdot \left( \frac{2 \cdot \lambda_S \cdot \Delta x}{\Delta y} \right) \cdot T_S^n$
Left wall	Value
$a_P$	$\frac{2 \cdot \beta \cdot \lambda_E \cdot \Delta y}{\Delta x} + \beta \cdot \alpha \cdot \Delta y$
$a_E$	$\frac{2 \cdot \beta \cdot \lambda_E \cdot \Delta y}{\Delta x}$
$a_W$	0
$a_N$	0
$a_S$	0
$b_P$	$\alpha \cdot T_g \cdot \Delta y + (1 - \beta) \cdot \left( -\alpha \cdot \Delta y - \frac{2 \cdot \lambda_E \cdot \Delta y}{\Delta x} \right) \cdot T_P^n + (1 - \beta) \cdot \left( \frac{2 \cdot \lambda_E \cdot \Delta y}{\Delta x} \right) \cdot T_E^n$
Right wall	Value
$a_P$	1
$a_E$	0
$a_W$	0
$a_N$	0
$a_S$	0
$b_P$	$8 + 0.005 \cdot t$

Table 6: A Two-Dimensional Transient Conduction Problem. Discretization coefficients of boundary nodes

Upper-left corner	Value	Upper-right corner	Value	Lower-left corner	Value	Lower-right corner	Value
$a_P$	1	$a_P$	1	$a_P$	1	$a_P$	1
$a_E$	0.5	$a_E$	0	$a_E$	0.5	$a_E$	0
$a_W$	0	$a_W$	0.5	$a_W$	0	$a_W$	0.5
$a_N$	0	$a_N$	0	$a_N$	0.5	$a_N$	0.5
$a_S$	0.5	$a_S$	0.5	$a_S$	0	$a_S$	0
$b_P$	0	$b_P$	0	$b_P$	0	$b_P$	0

Table 7: A Two-Dimensional Transient Conduction Problem. Discretization coefficients of corner nodes

Commenting the previous tables, the number “2” that appears in some coefficients of *Table 6* refers to the distance between nodes, which is half of the normal distance between inner nodes. Then, when dividing the temperature difference by the distance  $\Delta T/d_x$ , where  $d_x$  is  $\Delta x/2$ , it becomes  $2\Delta T/\Delta x$ . Note that the distance between nodes has been considered equal (except for the boundary nodes and their neighbors) to the distance between cell faces of a control volume due to the structured and equidistant mesh that has been implemented. The heat balance of the top cavity wall is shown in equation (53) and the heat balance regarding the left cavity wall is shown in equation (54).

$$\sum \dot{Q}_P = \frac{\lambda_S}{\frac{\Delta y}{2}} \cdot (T_S - T_P) \cdot \Delta x + \dot{Q}_{flow} \cdot \Delta x \quad (53)$$

$$\sum \dot{Q}_P = -\frac{\lambda_E}{\frac{\Delta x}{2}} \cdot (T_P - T_E) \cdot \Delta y - \alpha \cdot (T_P - T_g) \cdot \Delta y \quad (54)$$

In both cases, as these boundary nodes do not have associated control volumes, equation (48) is rewritten as follows:

$$0 = \beta \cdot \sum \dot{Q}_P^{n+1} + (1 - \beta) \cdot \sum \dot{Q}_P^n \quad (55)$$

Note that in the heat balance of equation (53) the heat flux corresponding to  $\dot{Q}_{flow}$  should be considered to be entering the control volume, as defined in the boundary conditions (*Table 5*). But in the developed code this heat flux was considered to be leaving the control volume. Then, in the code the sign of the flux must be changed to negative in order to solve this problem (see *Attachment 1*).

Regarding *Table 7*, the hypothesis of computing the variable at the corners using an arithmetic mean between the two neighbor nodes has been taken into account. This hypothesis is the most easy to be applied and does not change the global results into the domain, especially if the mesh is fine.

The inner nodes are discretized using a generic code as seen in *Table 8*, but a function which returns the physical properties of the control volume and its cell faces according to the material of the control volume itself and the material of the neighboring control volumes is used.

The heat balance of the inner nodes can be expressed as shown in equation (56) and, then, be applied to equation (48).

$$\begin{aligned} \sum \dot{Q}_P &= -\frac{\lambda_E}{\Delta x} \cdot (T_P - T_E) \cdot \Delta y + \frac{\lambda_W}{\Delta x} \cdot (T_W - T_P) \cdot \Delta y [...] \\ [...] &- \frac{\lambda_N}{\Delta y} \cdot (T_P - T_N) \cdot \Delta x + \frac{\lambda_S}{\Delta y} \cdot (T_S - T_P) \cdot \Delta x \end{aligned} \quad (56)$$

Inner nodes	Value
$a_P$	$\frac{\beta \cdot \lambda_E \cdot \Delta y}{\Delta x} + \frac{\beta \cdot \lambda_W \cdot \Delta y}{\Delta x} + \frac{\beta \cdot \lambda_N \cdot \Delta x}{\Delta y} + \frac{\beta \cdot \lambda_S \cdot \Delta x}{\Delta y} + \frac{\rho \cdot V \cdot c_P}{\Delta t}$
$a_E$	$\frac{\beta \cdot \lambda_E \cdot \Delta y}{\Delta x}$
$a_W$	$\frac{\beta \cdot \lambda_W \cdot \Delta y}{\Delta x}$
$a_N$	$\frac{\beta \cdot \lambda_N \cdot \Delta x}{\Delta y}$
$a_S$	$\frac{\beta \cdot \lambda_S \cdot \Delta x}{\Delta y}$
$b_P$	$\left( \frac{\rho \cdot V \cdot c_P}{\Delta t} + (1 - \beta) \cdot \left( -\frac{\lambda_E \cdot \Delta y}{\Delta x} - \frac{\lambda_W \cdot \Delta y}{\Delta x} - \frac{\lambda_S \cdot \Delta x}{\Delta y} - \frac{\lambda_N \cdot \Delta x}{\Delta y} \right) \right) \cdot T_P^n + (1 - \beta) \cdot \left( \frac{\lambda_E \cdot \Delta y}{\Delta x} \right) \cdot T_E^n + (1 - \beta) \cdot \left( \frac{\lambda_W \cdot \Delta y}{\Delta x} \right) \cdot T_W^n + (1 - \beta) \cdot \left( \frac{\lambda_S \cdot \Delta x}{\Delta y} \right) \cdot T_S^n + (1 - \beta) \cdot \left( \frac{\lambda_N \cdot \Delta x}{\Delta y} \right) \cdot T_N^n$

Table 8: A Two-Dimensional Transient Conduction Problem. Discretization coefficients of inner nodes

#### 2.4.1.4. Numerical resolution

Once the system of equations to be solved is known, the implementation of a proper solver is needed. There are some solvers that could be implemented in the code, like a simple Gauss-Seidel iterative solver, but in order to improve the performance of the code and taking advantage of the structured mesh that is going to be used on each problem of this study, finally a Line-by-Line method has been designed.

This method combines the Gauss-Seidel iterative process with the TDMA (Tri-Diagonal Matrix) algorithm. It consists in solving each row or column of the mesh using the direct solver of the TDMA algorithm until the system converges (this convergence is checked with Gauss-Seidel). In this case, the designed solver solves row by row.

The TDMA algorithm consists of rewriting the system of equations for each row as follows:

$$\phi[i] = P[i] \cdot \phi[i + 1] + R[i] \quad (57)$$

Therefore, equation (24) should be rewritten as:

$$a_P \phi_P^{n+1} = a_E \phi_E^{n+1} + a_W \phi_W^{n+1} + b_P' \quad (58)$$

Where  $b_P' = a_N \phi_N^{n+1} + a_S \phi_S^{n+1} + b_p$  and the values of the variable are the last known or the supposed ones (Gauss-Seidel method).

Combining equations (57) and (58), the coefficients  $P[i]$  and  $R[i]$  can be expressed, according to the discretization coefficients, as:

$$P[i] = \frac{a_E[i]}{a_P[i] - a_W[i] \cdot P[i-1]} \quad (59)$$

$$R[i] = \frac{b_P'[i] + a_W[i] \cdot R[i-1]}{a_P[i] - a_W[i] \cdot P[i-1]} \quad (60)$$

Then, the values of each row can be computed directly. First, the code must sweep from the first node (1) until the last one ( $N_x$ ) in the X direction of the row and the coefficients P and R can be computed easily using equations (59) and (60), with the following particularities:

$$P[1] = \frac{a_E[1]}{a_P[1]} \quad (61)$$

$$R[1] = \frac{b_P'[1]}{a_P[1]} \quad (62)$$

$$P[N_x] = 0 \quad (63)$$

$$R[N_x] = \frac{b_P'[N_x] + a_W[N_x] \cdot R[N_x-1]}{a_P[N_x] - a_W[N_x] \cdot P[N_x-1]} \quad (64)$$

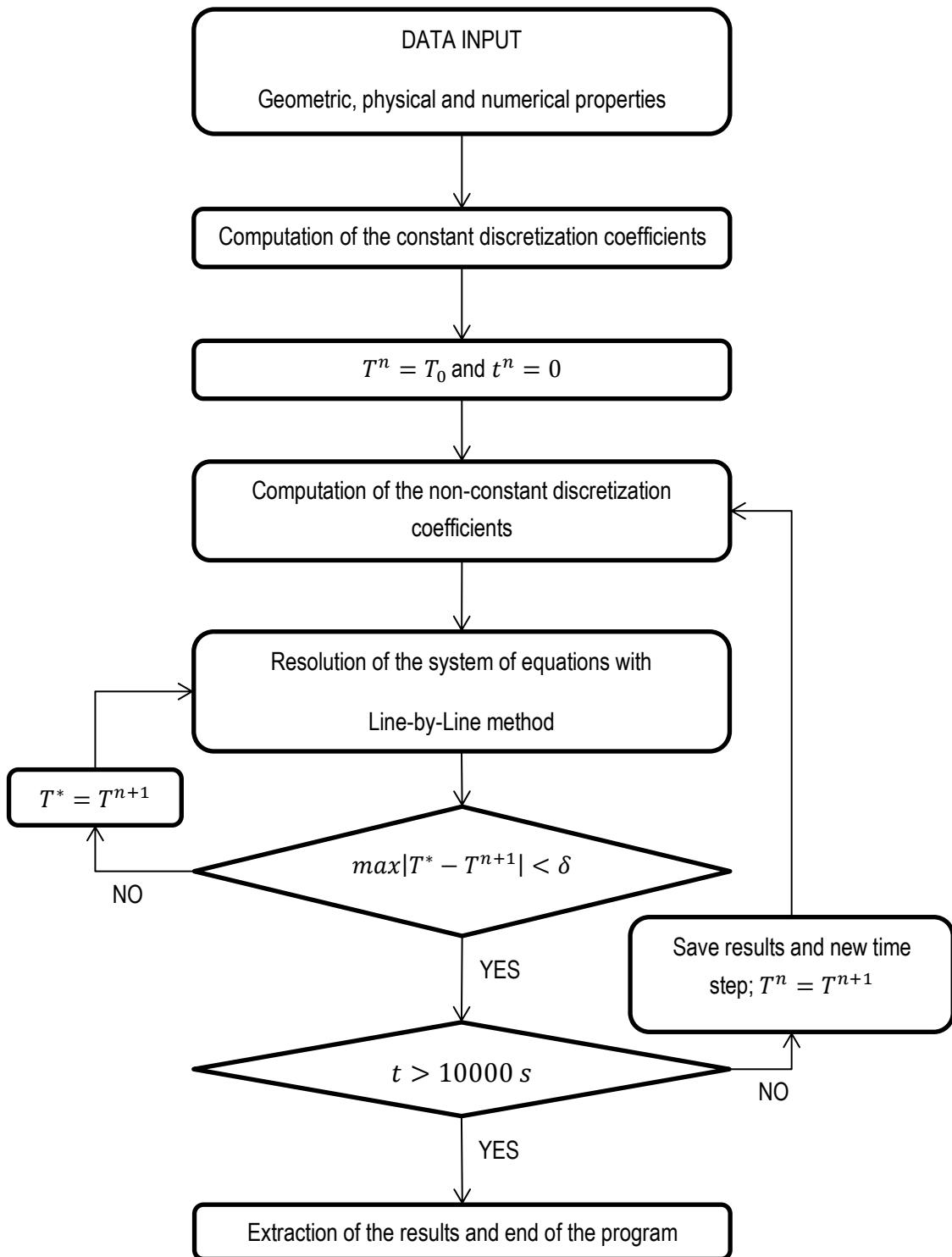
Once the coefficients are known, the values of the variable can be obtained sweeping from the last node to the first one using equation (57), with the following particularity:

$$\phi[N_x] = R[N_x] \quad (65)$$

The procedure now consists in changing from row to row repeating the same algorithm until the results converge.

When the entire domain is solved, the code calls a function that computes the value of the temperature at the desired points (remember that the code must give the temperature at locations (0.65, 0.56) and (0.74, 0.72)). This function uses the four nearest nodes of each desired point and computes its temperature, giving more importance to the temperature of the nodes according to the relative distance to the point.

#### 2.4.1.5. Global algorithm of resolution



#### 2.4.1.6. Results

Once the simulation is done, a file with the temperature at locations (0.65, 0.56) and (0.74, 0.72) for each time step until 10000 seconds is obtained. It is also possible to extract the temperature on the whole domain in order to make a contour plot and have an overall idea about what is happening in the rod.

Some of the results for different instants are shown in *Table 9*.

Time [s]	Location 1 (0.65, 0.56) Temperature (°C)	Location 2 (0.74, 0.72) Temperature (°C)	Time [s]	Location 1 (0.65, 0.56) Temperature (°C)	Location 2 (0.74, 0.72) Temperature (°C)
0	<b>8.00000</b>	<b>8.00000</b>	3353	<b>20.1907</b>	<b>20.1912</b>
5	8.00000	8.00001	3400	20.3234	20.3503
10	8.00000	8.00012	3800	21.4269	21.6795
20	8.00000	8.00095	4200	22.4914	22.9714
50	8.00013	8.00552	4600	23.5259	24.2344
100	8.00545	8.01435	4800	24.0340	24.8571
150	8.04900	8.02624	<b>5000</b>	<b>24.5371</b>	<b>25.4750</b>
200	8.16219	8.04917	5200	25.0356	26.0885
300	8.56037	8.16141	5500	25.7761	27.0017
400	9.06389	8.37920	6000	26.9945	28.5087
500	9.58777	8.68787	6300	27.7181	29.4058
600	10.1031	9.06212	6600	28.4373	30.2986
700	10.6029	9.47941	6900	29.1528	31.1878
800	11.0867	9.92275	7200	29.8653	32.0742
900	11.5554	10.3802	7500	30.5754	32.9582
1000	12.0103	10.8436	7800	31.2834	33.8403
1100	12.4524	11.3077	8000	31.7544	34.4274
1200	12.8826	11.7690	8200	32.2248	35.0138
1300	13.3015	12.2253	8400	32.6946	35.5997
1400	13.7097	12.6754	8600	33.1639	36.1852
1500	14.1077	13.1185	8800	33.6327	36.7701
1800	15.2462	14.4035	9000	34.1011	37.3548
2100	16.3109	15.6239	9200	34.5692	37.9390
2400	17.3133	16.7869	9400	35.0370	38.5230
2700	18.2633	17.9007	9600	35.5045	39.1067
3000	19.1698	18.9731	9800	35.9718	39.6902
<b>3352</b>	<b>20.1879</b>	<b>20.1878</b>	<b>10000</b>	<b>36.4389</b>	<b>40.2735</b>

Table 9: A Two-Dimensional Transient Conduction Problem. Some results for a 165x120 mesh and  $\beta = 0.5$

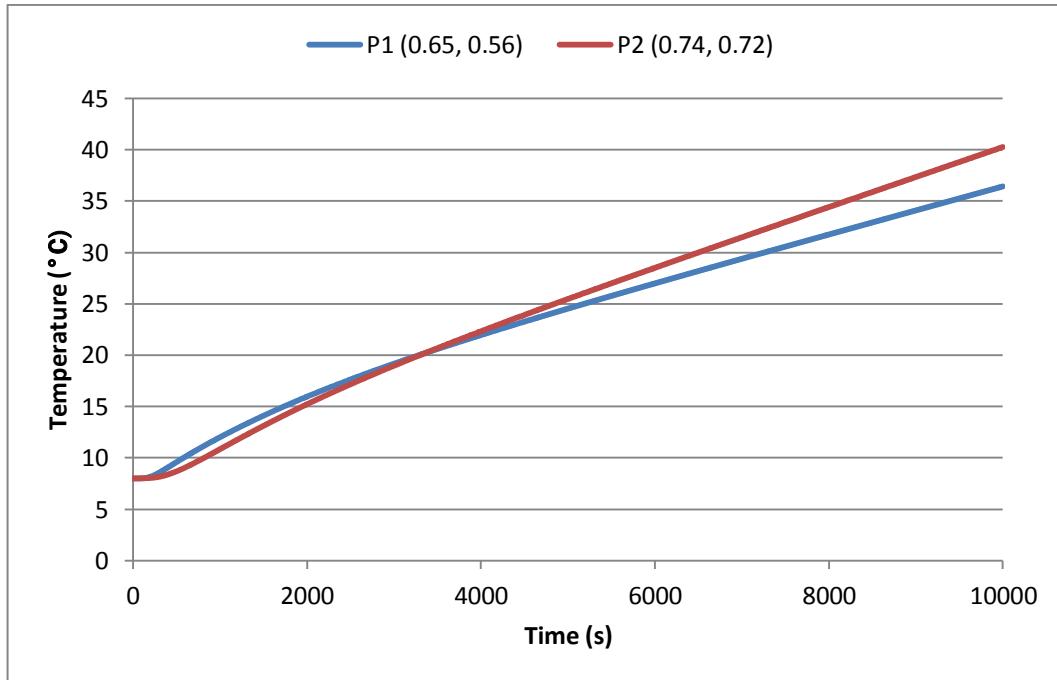


Figure 7: A Two-Dimensional Transient Conduction Problem. Evolution of temperature with time for a 165x120 mesh, time step 1 s and  $\beta = 0.5$

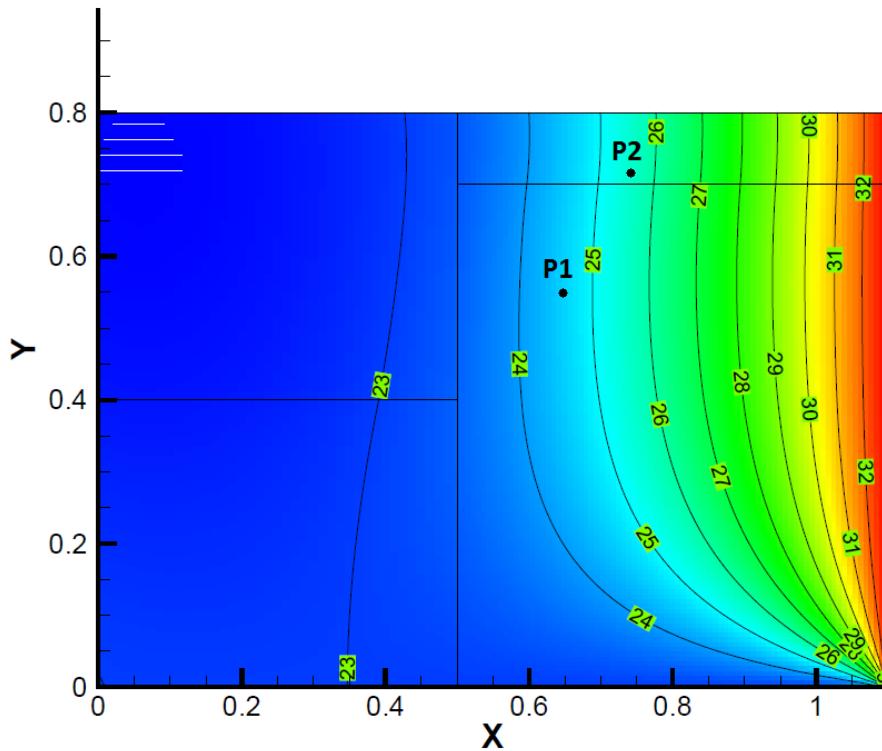


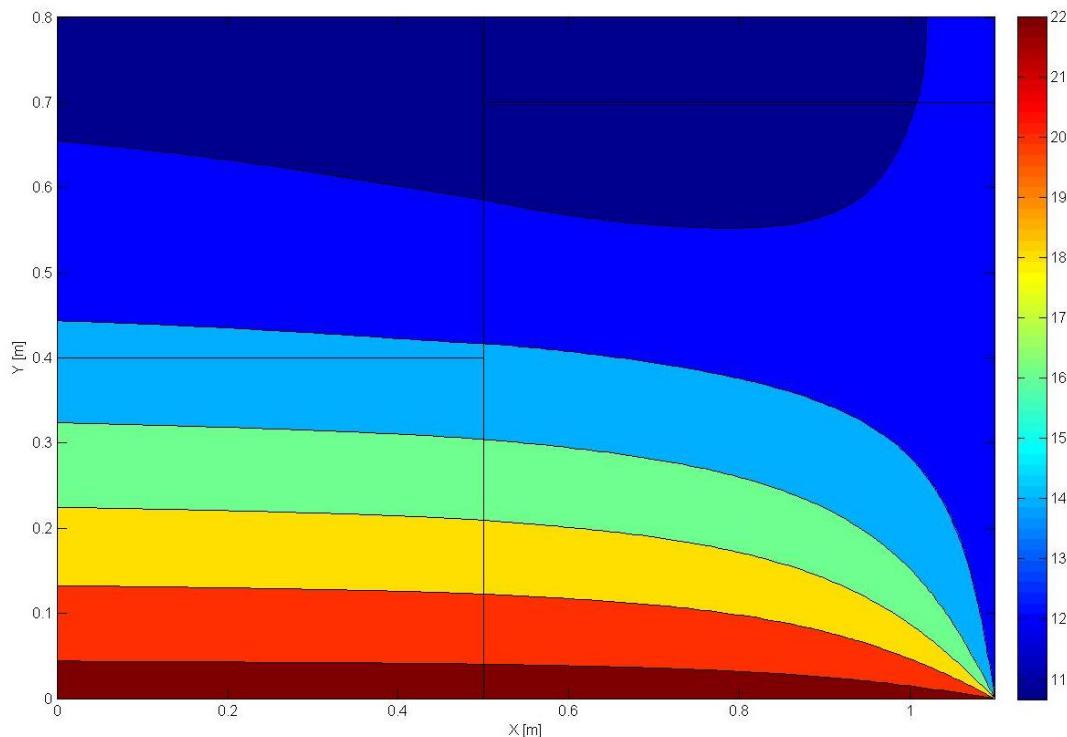
Figure 8: A Two-Dimensional Transient Conduction Problem. Reference solution for  $t = 5000$  s

The variation of the values of the temperature shown in *Table 9* can be observed in *Figure 7*. At the first instance, the temperature in P1 increases faster than in P2, but, then, this situation changes

and, as seen in *Table 9*, at the instant  $t = 3353\text{ s}$  the temperature in P2 becomes bigger than the temperature in P1. This phenomenon occurs because at the first instants the boundary condition of the bottom wall of the cavity is stronger than the other boundary conditions (see *Figure 9* and *Figure 10*). The point P1 is closer to the bottom of the cavity than the point P2 and, therefore, its temperature increases faster than the temperature of P2 at the very first beginning. After that, when the temperature of the right cavity wall becomes enough important because of the simulation time, the temperature of point P2 starts to increase faster than the temperature in P1. The difference between the temperature increase ratios of the two points becomes bigger with time.

Analyzing the obtained solution, according to the results for the instant  $t = 5000\text{ s}$  ( $T_{P1} = 24.5371^\circ\text{C}$  and  $T_{P2} = 25.4750^\circ\text{C}$ ), the temperature of the points yields inside the proper temperature range for each one. The temperature of P1 must be between  $24^\circ\text{C}$  and  $25^\circ\text{C}$  and the temperature of P2 must be between  $25^\circ\text{C}$  and  $26^\circ\text{C}$ . Furthermore, observing *Figure 8*, where the approximate position of the points has been marked, the temperature of P1 should be around  $24.5^\circ\text{C}$  and the temperature of P2 around  $25.5^\circ\text{C}$ . The obtained results meet the reference solution in a proper way and this fact could be used in order to validate and verify the developed code.

The following figures show the temperature field for the instants  $t = 1000\text{ s}$ ,  $t = 5000\text{ s}$  and  $t = 10000\text{ s}$ .



**Figure 9:** A Two-Dimensional Transient Conduction Problem. Instantaneous isotherms at  $t = 1000\text{ s}$  for a  $165 \times 120$  mesh, time step  $1\text{ s}$  and  $\beta = 0.5$  (temperature in  $^\circ\text{C}$ )

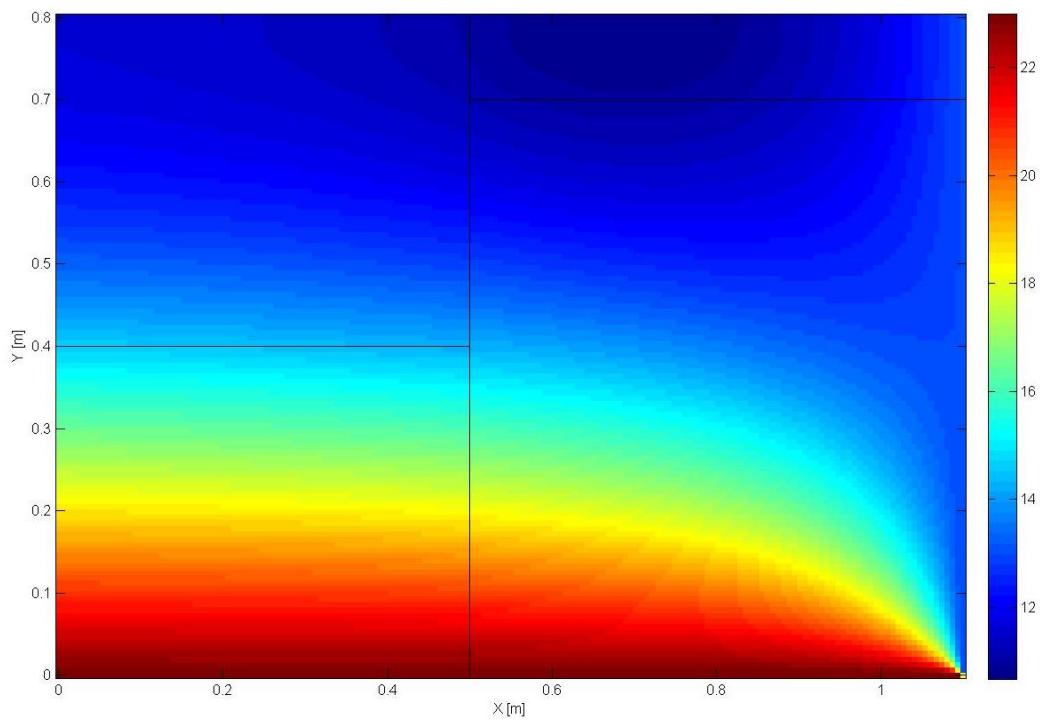


Figure 10: A Two-Dimensional Transient Conduction Problem. Temperature distribution at  $t = 1000$  s for a 165x120 mesh, time step 1 s and  $\beta = 0.5$  (temperature in  $^{\circ}\text{C}$ )

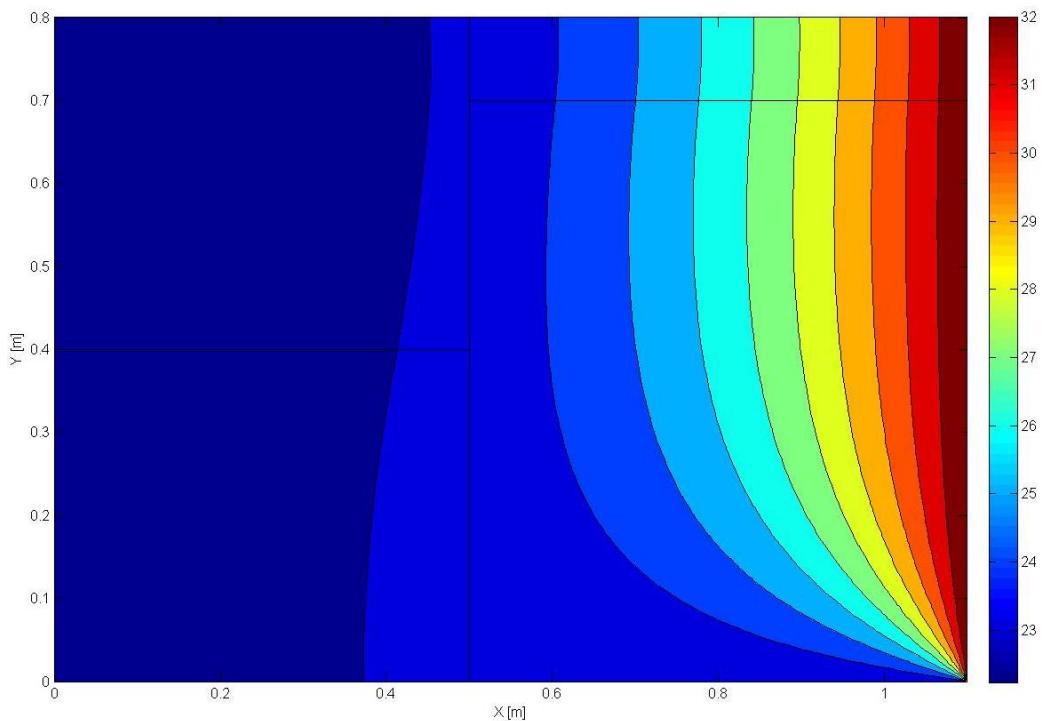


Figure 11: A Two-Dimensional Transient Conduction Problem. Instantaneous isotherms at  $t = 5000$  s for a 165x120 mesh, time step 1 s and  $\beta = 0.5$  (temperature in  $^{\circ}\text{C}$ )

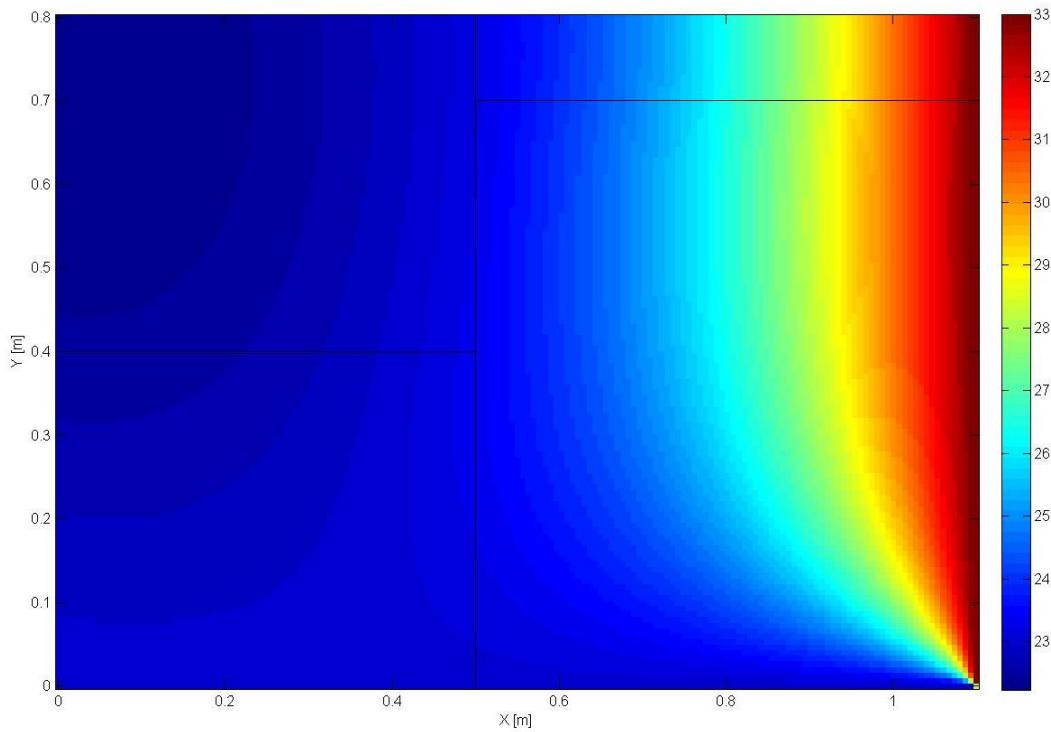


Figure 12: A Two-Dimensional Transient Conduction Problem. Temperature distribution at  $t = 5000$  s for a 165x120 mesh, time step 1 s and  $\beta = 0.5$  (temperature in  $^{\circ}\text{C}$ )

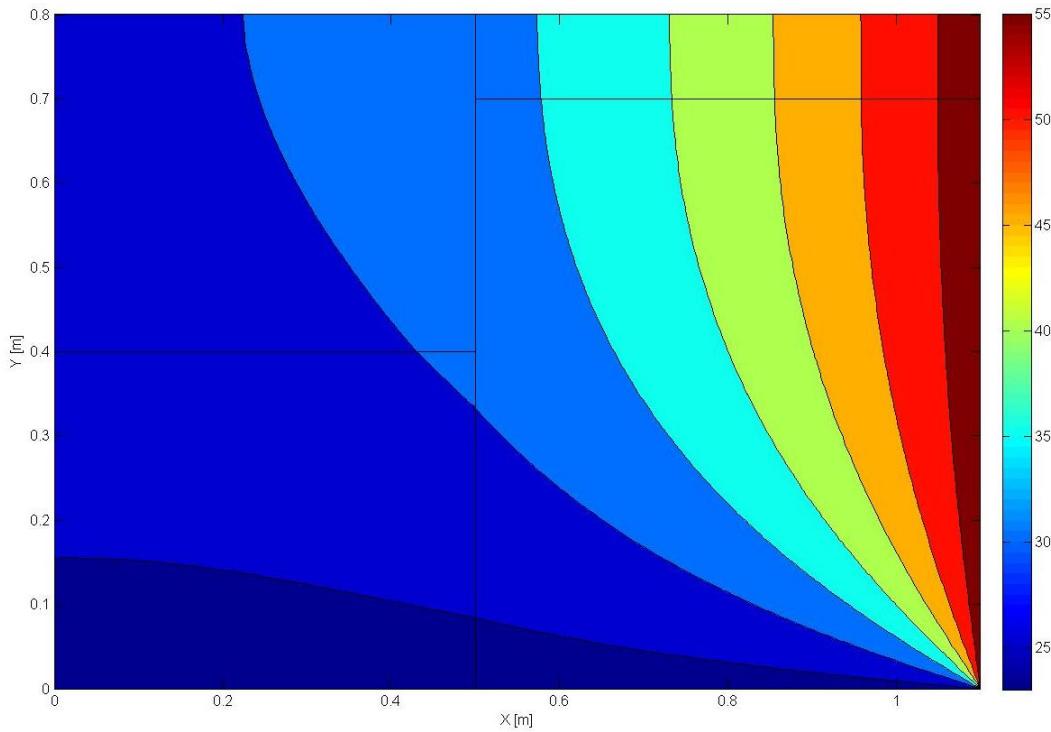


Figure 13: A Two-Dimensional Transient Conduction Problem. Instantaneous isotherms at  $t = 10000$  s for a 165x120 mesh, time step 1 s and  $\beta = 0.5$  (temperature in  $^{\circ}\text{C}$ )

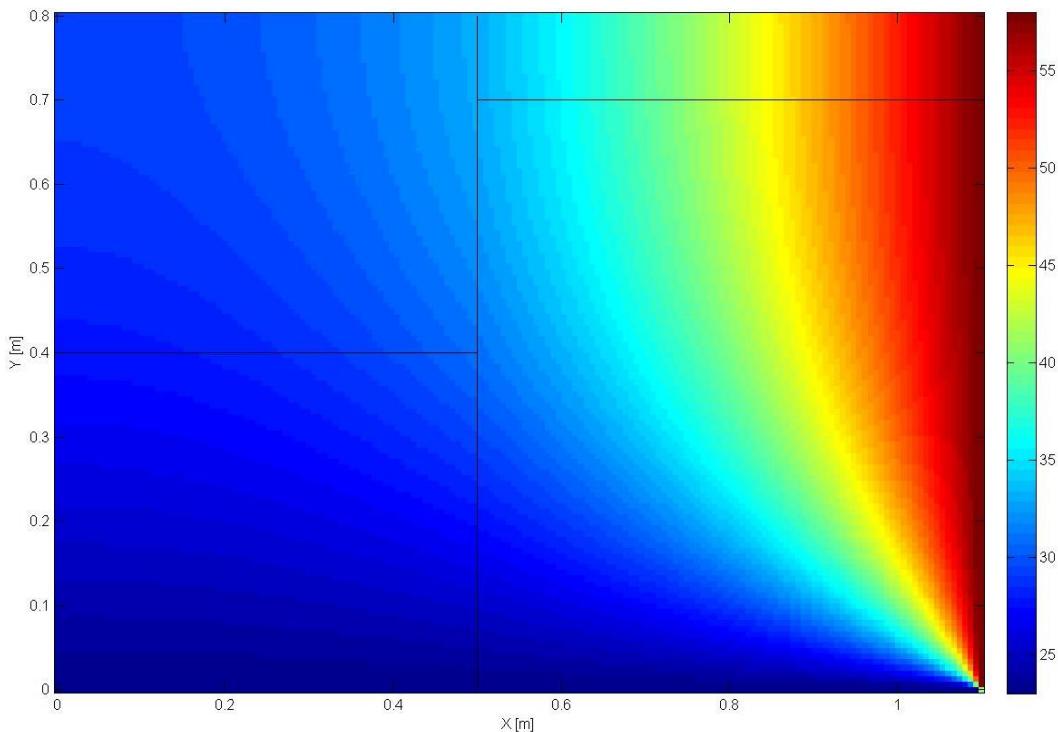


Figure 14: A Two-Dimensional Transient Conduction Problem. Temperature distribution at  $t = 10000$  s for a 165x120 mesh, time step 1 s and  $\beta = 0.5$  (temperature in °C)

#### 2.4.1.7. Conclusions

The first point to comment in this section is the consistency of the obtained results and the simulation. The results show the good implementation of the boundary conditions into the code, since the evolution of the temperature with time corresponds to the expected evolution according to the boundary and initial conditions of the case.

Once the contour conditions are seen to be well implemented, it is important to discuss if the discretization coefficients of each control volume has been good implemented too. In this case, the similitude between *Figure 8* and *Figure 12* and the obtained results for the two studied points show that the inner discretization seems to be well implemented.

At this point, one interesting study to be done with the developed code would be the quickness in convergence according to the temporal discretization. As the program allows the user to change the parameter  $\beta$ , a fully explicit or a fully implicit scheme could be implemented. The displayed results have been obtained for a  $\beta = 0.5$  parameter, that follows a Crank-Nicolson temporal scheme.

In general, for every  $\beta$  parameter bigger than zero, the system of equations to be solved becomes harder. This happens because for  $\beta$  different to zero, the equation for each node takes into account the values of the future variables of the surrounding nodes, requiring more computational efforts to solve the system of equations. However, with  $\beta = 0$  (fully explicit scheme) the system of equations would be easier to solve since the temperature at each node would only depend on the

temperature of the surrounding nodes at the past instant, which are already known. Nevertheless, the convergence of this temporal scheme could not be assured.

#### 2.4.2. The Smith-Hutton Problem

##### 2.4.2.1. Objective

This is another proposed problem by CTTC as an application of the convection-diffusion equation in steady form. The objective of the problem is to solve the transport of a variable due to a constant velocity field into a bi-dimensional body testing different low-order numerical schemes.

##### 2.4.2.2. Problem definition

The problem consists in a bi-dimensional body with a prescribed velocity field. A variable is transported from an inlet to an outlet with different boundary conditions. The situation must be treated as a 2-D problem and the required solution corresponds to the steady state.

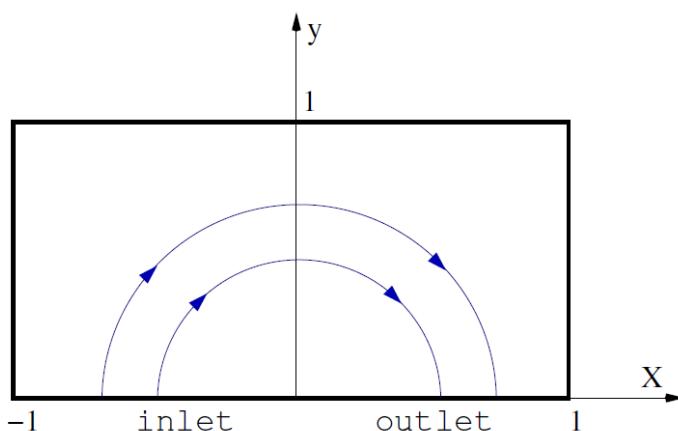


Figure 15: The Smith-Hutton Problem. Schema of the proposed problem (by CTTC)

The prescribed velocity field corresponds to:

$$u(x, y) = 2y \cdot (1 - x^2) \quad (66)$$

$$v(x, y) = -2x \cdot (1 - y^2) \quad (67)$$

And the boundary conditions are:

$$\phi = 1 + \tanh(\alpha(2x + 1)) \rightarrow y = 0; x \in (-1, 0) \text{ (inlet)} \quad (68)$$

$$\frac{\partial \phi}{\partial y} = 0 \rightarrow y = 0; x \in (0, 1) \text{ (outlet)} \quad (69)$$

$$\phi = 1 - \tanh(\alpha) \text{ (elsewhere)} \quad (70)$$

Where  $\alpha = 10$ .

Remembering the convection-diffusion equation to be solved (see equation (71)), the following cases must be solved:

$\rho/\Gamma$
10
$10^3$
$10^6$

Table 10: The Smith-Hutton Problem. Cases to be solved

$$\frac{\partial \rho \phi}{\partial t} + \nabla \cdot (\rho \vec{v} \phi) = \nabla \cdot (\Gamma \nabla \phi) + S \quad (71)$$

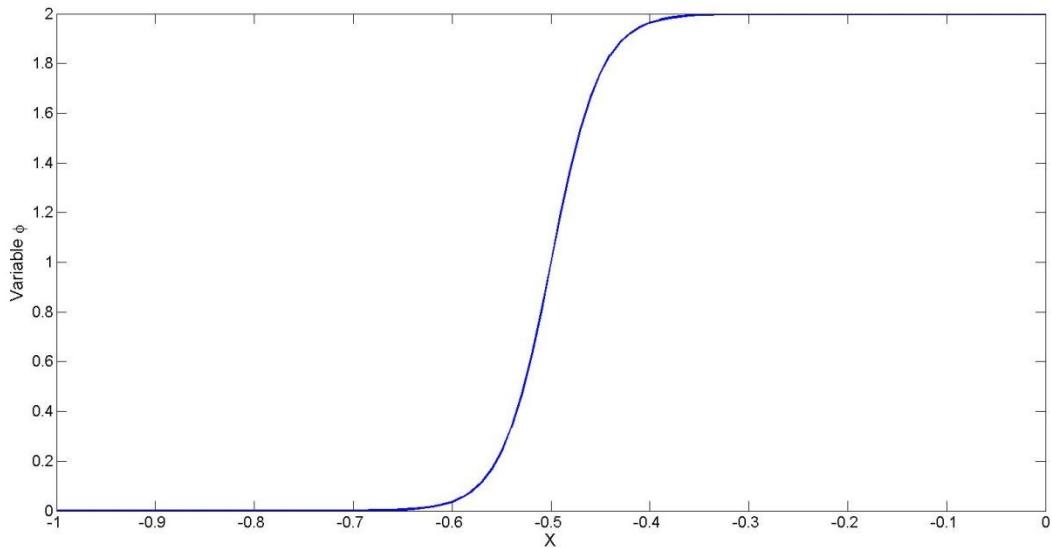


Figure 16: The Smith-Hutton Problem. Value of  $\phi$  at the inlet (equation (68))

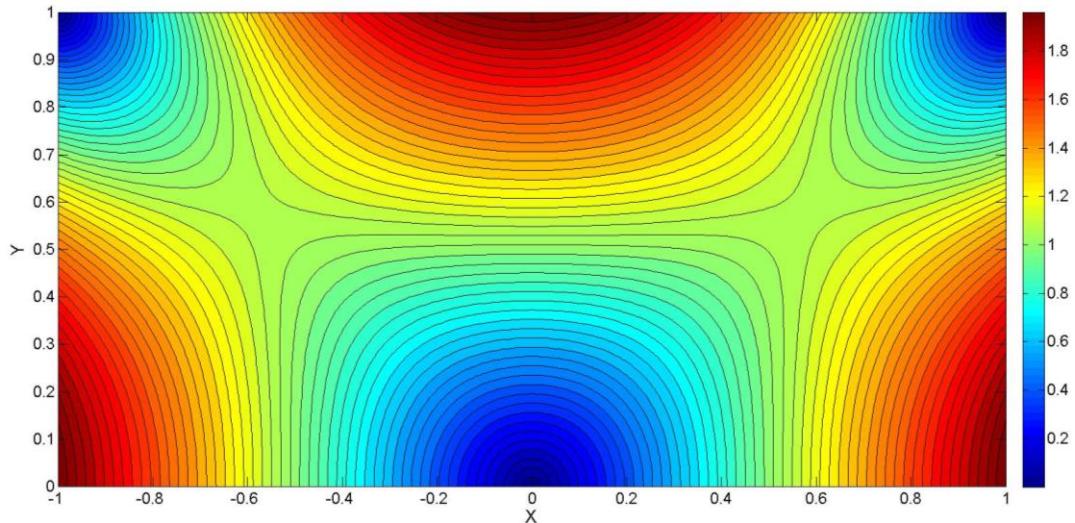


Figure 17: The Smith-Hutton Problem. Isolines of the prescribed velocity field

The boundary condition at the inlet can be seen in *Figure 16*. This type of contour condition is problematic because of the quick change in its value. It can be seen that the variable goes from zero to two in about 0.2 units and, therefore, the mesh should be fine enough in order to capture

the variations of the variable. This fact is reflected in the values of the variable at the outlet as it will be seen in further sections.

#### 2.4.2.3. Numerical discretization

The convection-diffusion equation to be discretized is equation (71), but as the physical properties are supposed to be constant in the entire domain, the previous equation can be rewritten as follows, in order to be able to change the parameter  $\rho/\Gamma$  in an easier way, and taking into account that no source term exists in this problem.

$$\frac{\rho}{\Gamma} \frac{\partial \phi}{\partial t} + \frac{\rho}{\Gamma} \nabla \cdot (\vec{v}\phi) = \nabla \cdot (\nabla \phi) \quad (72)$$

The previous equation, once discretized (see equation (11)), is written as:

$$\begin{aligned} & \frac{\rho}{\Gamma} \frac{(\phi)_P^{n+1} - (\phi)_P^n}{\Delta t} \Delta x \Delta y + \frac{\rho}{\Gamma} [(u\phi)_e^{n+1} - (u\phi)_w^{n+1}] \cdot \Delta y + \\ & + \frac{\rho}{\Gamma} [(v\phi)_n^{n+1} - (v\phi)_s^{n+1}] \cdot \Delta x = \left[ \left( \frac{\partial \phi}{\partial x} \right)_e^{n+1} - \left( \frac{\partial \phi}{\partial x} \right)_w^{n+1} \right] \cdot \Delta y + \\ & + \left[ \left( \frac{\partial \phi}{\partial y} \right)_n^{n+1} - \left( \frac{\partial \phi}{\partial y} \right)_s^{n+1} \right] \cdot \Delta x \end{aligned} \quad (73)$$

The mesh used is the same mesh as *Figure 6* and a system of equations like equation (24) must be obtained. Then, the discretization coefficients for the boundary and corner nodes can be computed. Note that the discretization of the corner nodes is the same as the previous studied case and, as it was said, in all the problems this hypothesis for the discretization has been chosen.

Inlet	Value	Outlet	Value	Everywhere else	Value
$a_P$	1	$a_P$	1	$a_P$	1
$a_E$	0	$a_E$	0	$a_E$	0
$a_W$	0	$a_W$	0	$a_W$	0
$a_N$	0	$a_N$	1	$a_N$	0
$a_S$	0	$a_S$	0	$a_S$	0
$b_P$	$1 + \tanh(\alpha(2x + 1))$	$b_P$	0	$b_P$	$1 - \tanh(\alpha)$

Table 11: The Smith-Hutton Problem. Discretization coefficients of boundary nodes

Upper-left corner	Value	Upper-right corner	Value	Lower-left corner	Value	Lower-right corner	Value
$a_P$	1	$a_P$	1	$a_P$	1	$a_P$	1
$a_E$	0.5	$a_E$	0	$a_E$	0.5	$a_E$	0
$a_W$	0	$a_W$	0.5	$a_W$	0	$a_W$	0.5
$a_N$	0	$a_N$	0	$a_N$	0.5	$a_N$	0.5
$a_S$	0.5	$a_S$	0.5	$a_S$	0	$a_S$	0
$b_P$	0	$b_P$	0	$b_P$	0	$b_P$	0

Table 12: The Smith-Hutton Problem. Discretization coefficients of corner nodes

For the inner nodes, the type of discretization follows the guides of section 2.3., but with a few changes as the discretized equation is a little bit different. Then,

$$a_N = D_n \cdot A(|P_n|) + \max(-F_n, 0) \quad (74)$$

$$a_S = D_s \cdot A(|P_s|) + \max(F_s, 0) \quad (75)$$

$$a_E = D_e \cdot A(|P_e|) + \max(-F_e, 0) \quad (76)$$

$$a_W = D_w \cdot A(|P_w|) + \max(F_w, 0) \quad (77)$$

$$a_P = a_N + a_S + a_E + a_W + \frac{\rho}{\Gamma} \cdot \frac{\Delta x \Delta y}{\Delta t} \quad (78)$$

$$b_P = \frac{\rho}{\Gamma} \cdot \frac{\Delta x \Delta y}{\Delta t} \cdot \phi_P^n \quad (79)$$

Where,

$$D_n = \frac{\Delta x}{(\delta y)_n}; D_s = \frac{\Delta x}{(\delta y)_s}; D_e = \frac{\Delta y}{(\delta x)_e}; D_w = \frac{\Delta y}{(\delta x)_w} \quad (80)$$

$$F_n = \frac{\rho}{\Gamma} (v)_n \Delta x; F_s = \frac{\rho}{\Gamma} (v)_s \Delta x; F_e = \frac{\rho}{\Gamma} (u)_e \Delta y; F_w = \frac{\rho}{\Gamma} (u)_w \Delta y \quad (81)$$

The developed code only applies for low order numerical schemes. Because of this, the value of  $A(|P|)$  is taken from *Table 2*.

Since the desired results are the ones for the steady state of the problem, the code could be simplified eliminating all the terms that have time step contributions of the discretization coefficients. With this simplification, the solution would be obtained quickly, but no study of the proper needed time step for each case could be done. Therefore, the results shown in the following sections have been obtained with this simplification, but the two different codes according to each case are provided (see *Attachment 1*).

#### 2.4.2.4. Numerical resolution

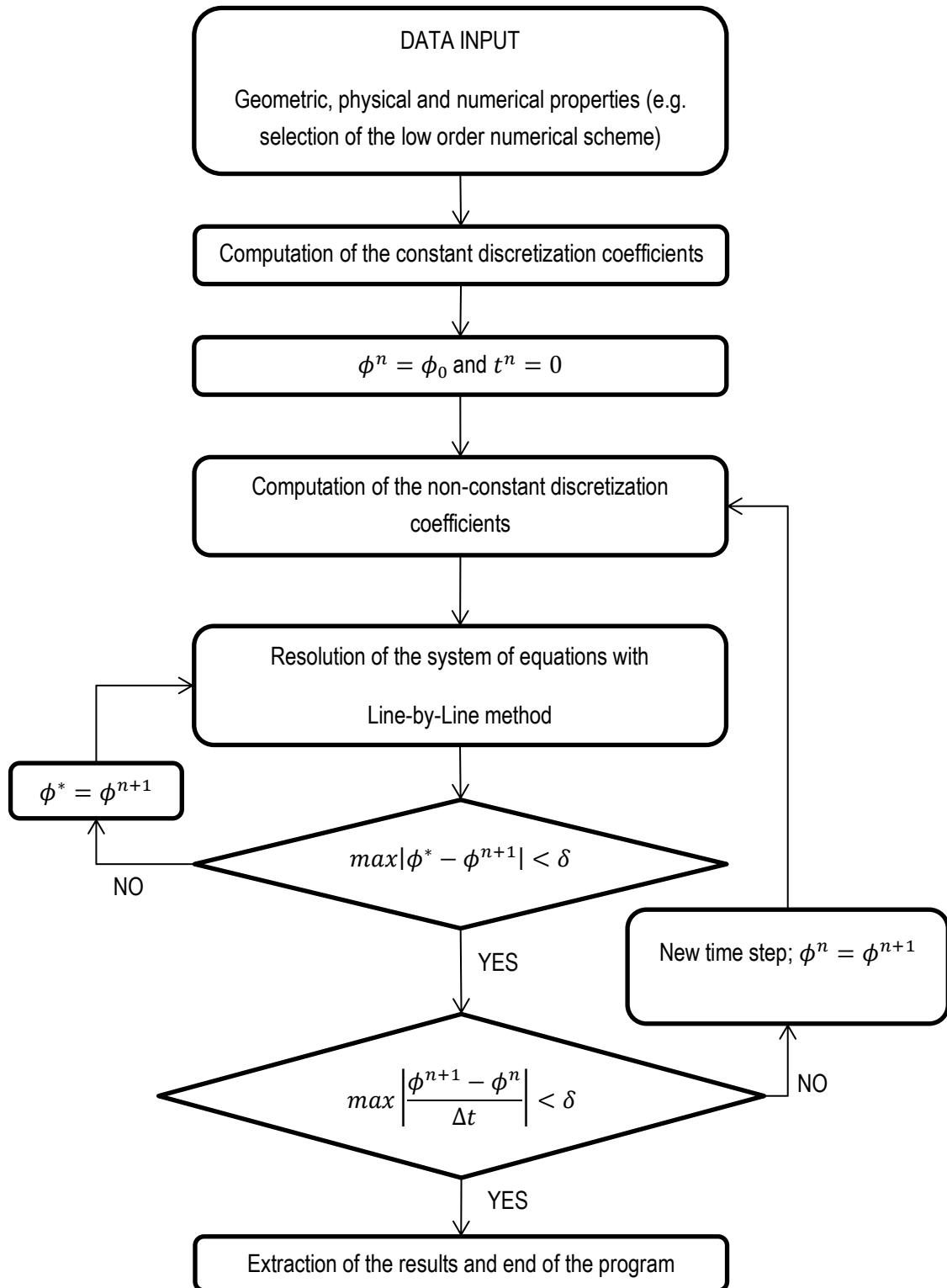
The solver chosen to solve the system of equations is again the Line-By-Line method seen in section 2.4.1.4.

Furthermore, in this section it is important to discuss when the solution according to the first proposed method is steady (transient method). It could be chosen the option to compare just the variables between the new time step and the previous one until some convergence criteria is achieved. But as the solution depends on the time step, it is better to compare the variables taking into account the time step used to compute the new value of the variable (this is done in the other problems too). Then, the equation that expresses the convergence criterion is equation (82).

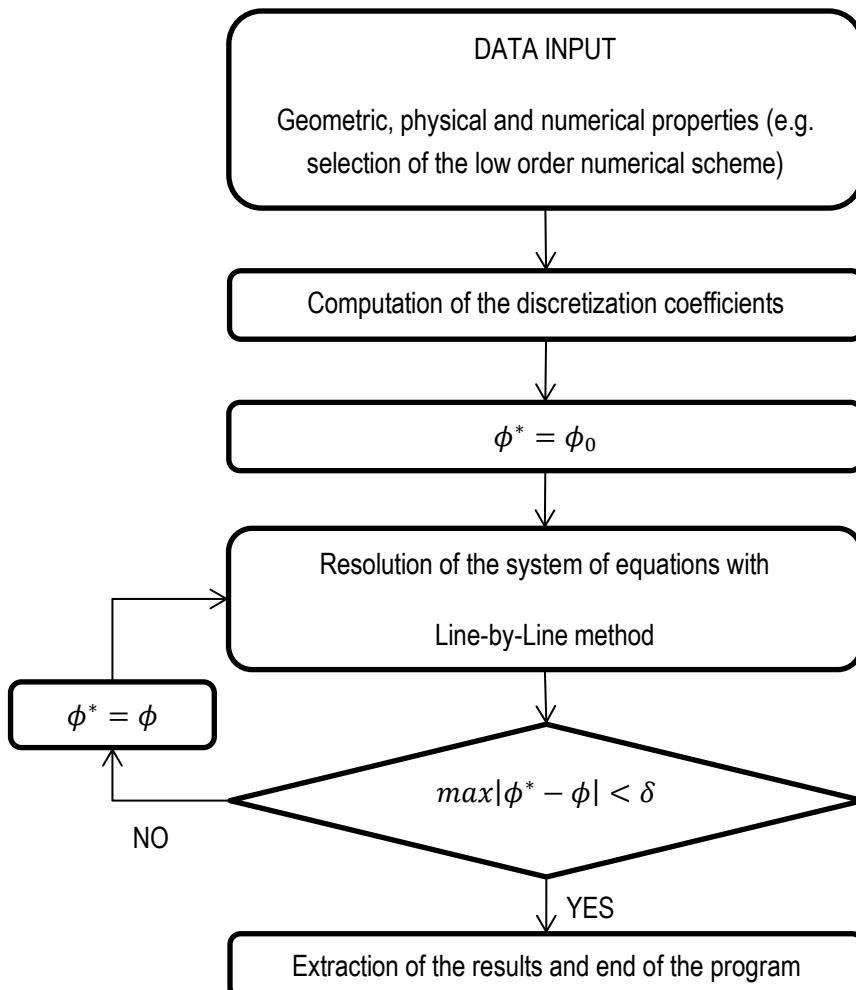
$$\frac{\phi^{n+1} - \phi^n}{\Delta t} < \delta \quad (82)$$

#### 2.4.2.5. Global algorithm of resolution

##### 2.4.2.5.1. Transient method



#### 2.4.2.5.2. Steady method



#### 2.4.2.6. Results

The reference results for this problem are shown in the following table:

X-Coordinate	$\frac{\rho}{\Gamma} = 10$	$\frac{\rho}{\Gamma} = 10^3$	$\frac{\rho}{\Gamma} = 10^6$
0.0	1.989	2.0000	2.000
0.1	1.402	1.9990	2.000
0.2	1.146	1.9997	2.000
0.3	0.946	1.9850	1.999
0.4	0.775	1.8410	1.964
0.5	0.621	0.9510	1.000
0.6	0.480	0.1540	0.036
0.7	0.349	0.0010	0.001
0.8	0.227	0.0000	0.000
0.9	0.111	0.0000	0.000
1.0	0.000	0.0000	0.000

Table 13: The Smith-Hutton Problem. Reference solution

The results obtained with UDS scheme are shown in the following pages. Furthermore, a comparison between different numerical schemes for a specific case is done. In order to read all the results obtained with the different low numerical schemes, refer to *Attachment 2*.

Case  $\rho/\Gamma = 10$

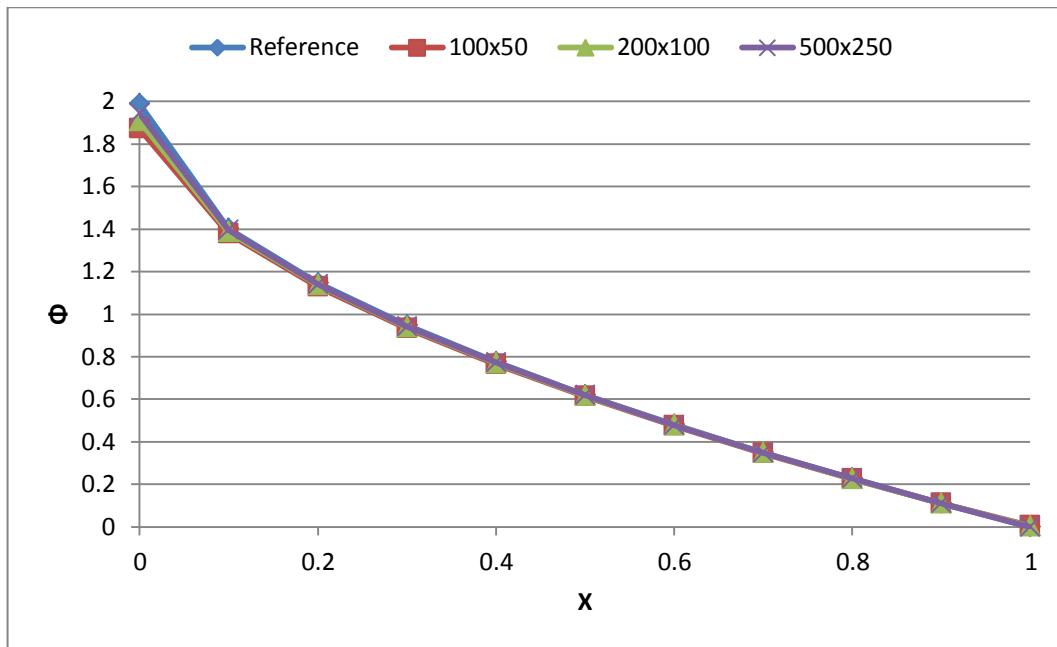


Figure 18: The Smith-Hutton Problem. Comparison of results between different mesh sizes using UDS scheme and  $\rho/\Gamma = 10$  (1)

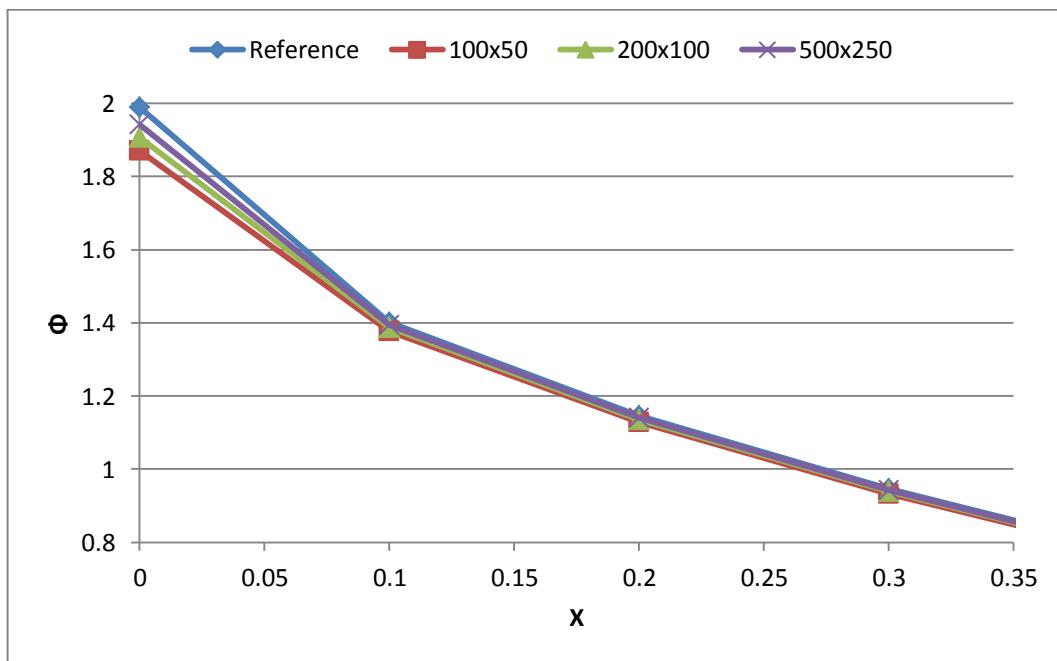
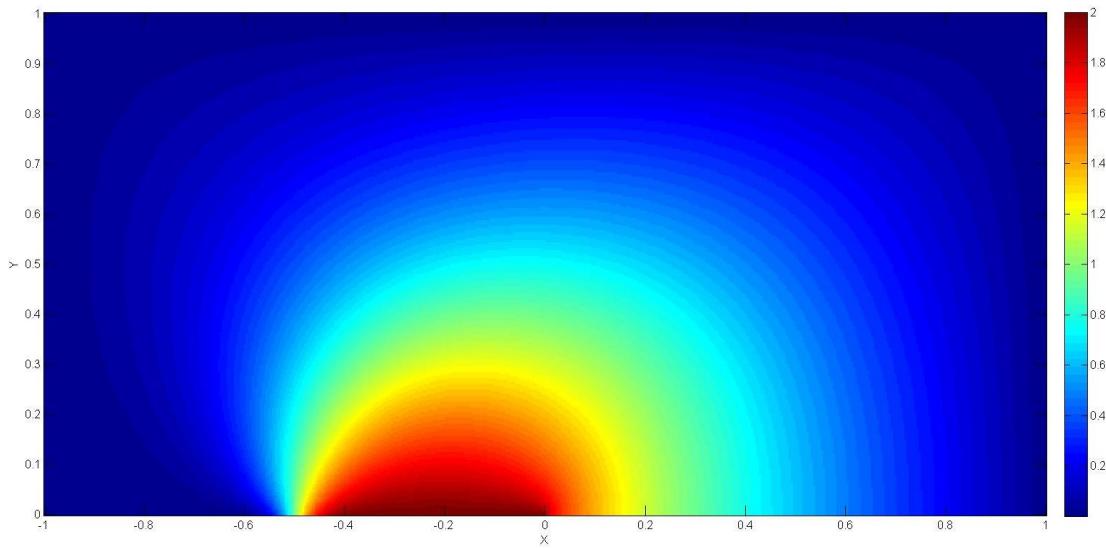
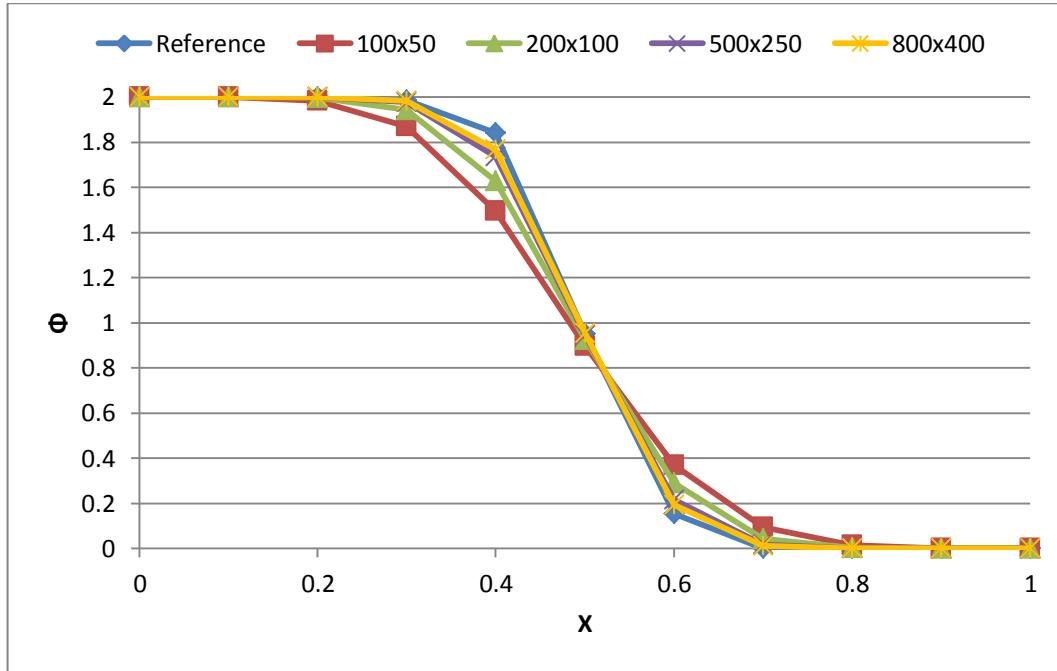


Figure 19: The Smith-Hutton Problem. Comparison of results between different mesh sizes using UDS scheme and  $\rho/\Gamma = 10$  (2)

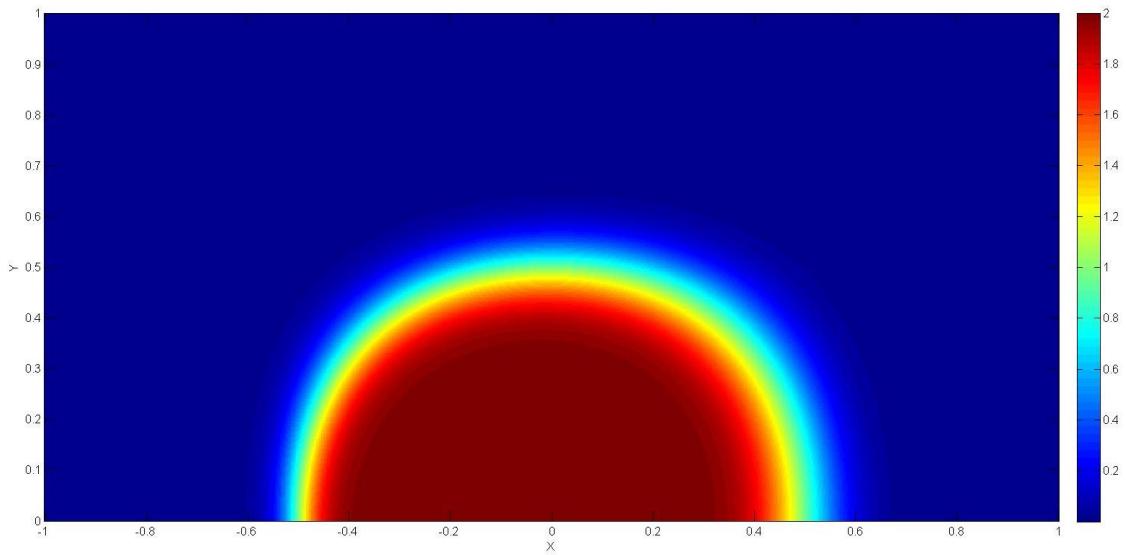


**Figure 20:** The Smith-Hutton Problem. Distribution of  $\phi$  in the domain using a 500x250 mesh, UDS scheme and  $\rho/\Gamma = 10$

Case  $\rho/\Gamma = 10^3$

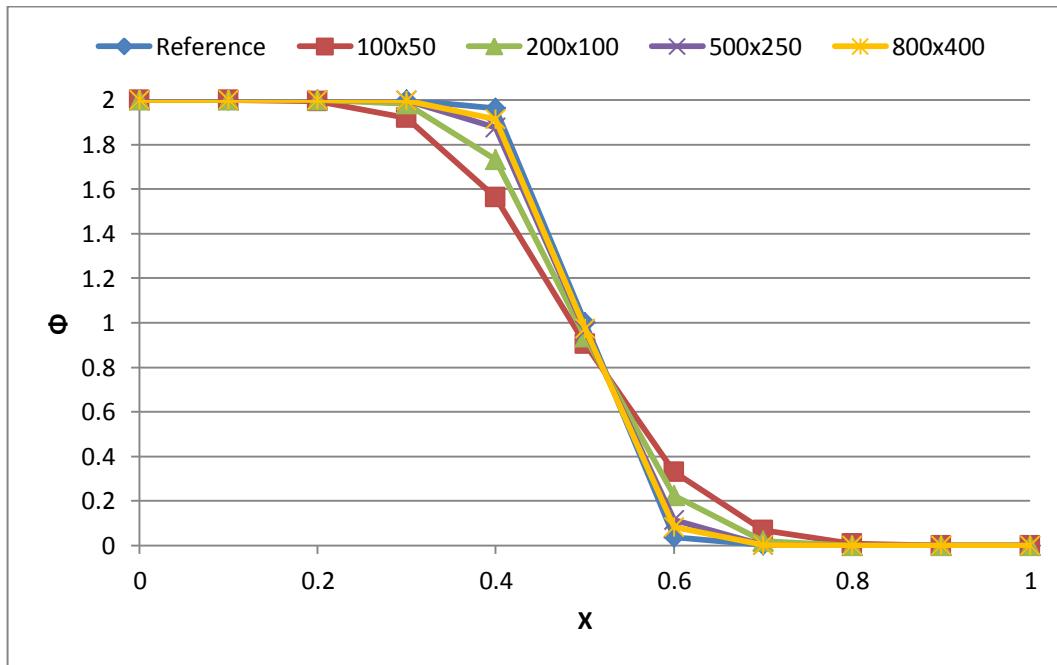


**Figure 21:** The Smith-Hutton Problem. Comparison of results between different mesh sizes using UDS scheme and  $\rho/\Gamma = 10^3$

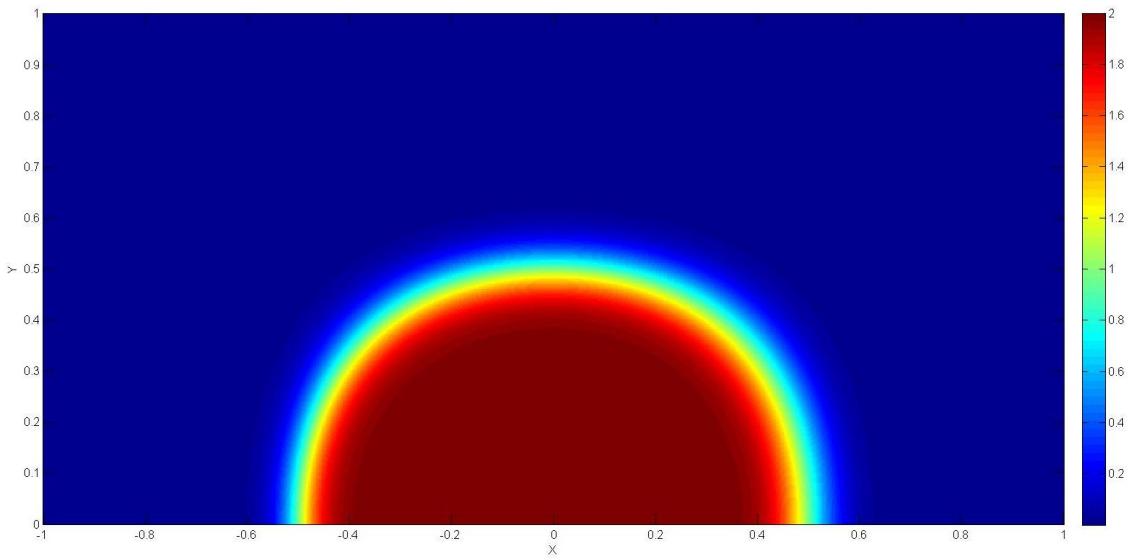


**Figure 22:** The Smith-Hutton Problem. Distribution of  $\phi$  in the domain using a 800x400 mesh, UDS scheme and  $\rho/\Gamma = 10^3$

Case  $\rho/\Gamma = 10^6$

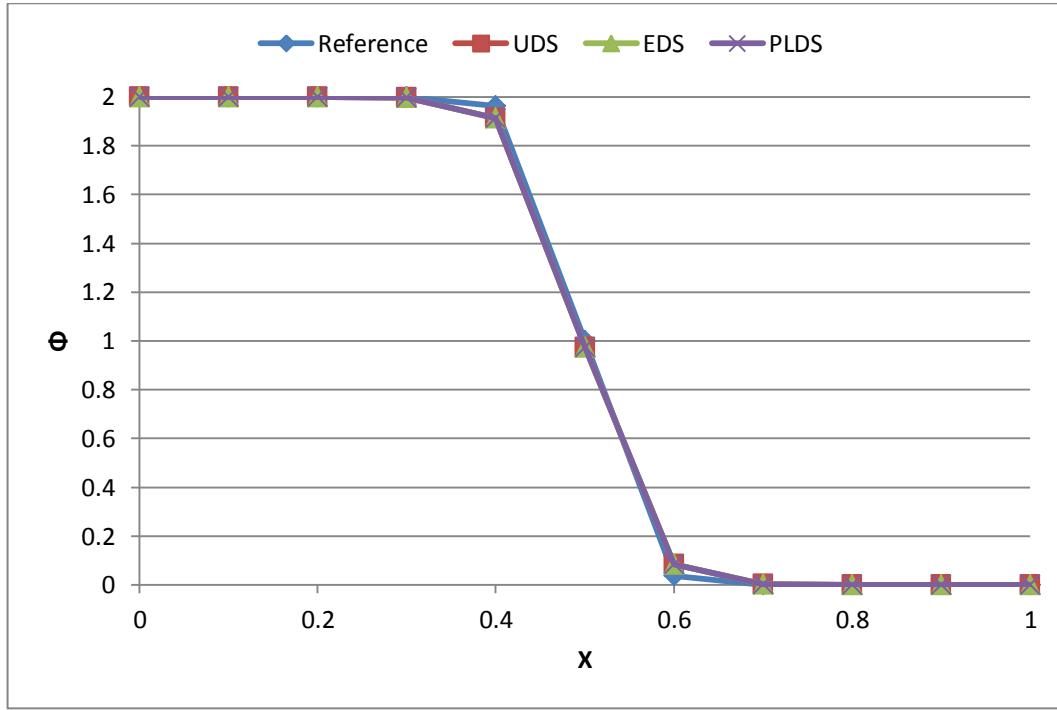


**Figure 23:** The Smith-Hutton Problem. Comparison of results between different mesh sizes using UDS scheme and  $\rho/\Gamma = 10^6$



**Figure 24:** The Smith-Hutton Problem. Distribution of  $\phi$  in the domain using a 800x400 mesh, UDS scheme and  $\rho/\Gamma = 10^6$

Now, a comparison between different numerical schemes is done for the case of  $\rho/\Gamma = 10^6$  and the case of  $\rho/\Gamma = 10^3$  for the mesh of 800x400. The results will be commented in the following section 2.4.2.7.



**Figure 25:** The Smith-Hutton Problem. Comparison between different numerical schemes  $\rho/\Gamma = 10^6$

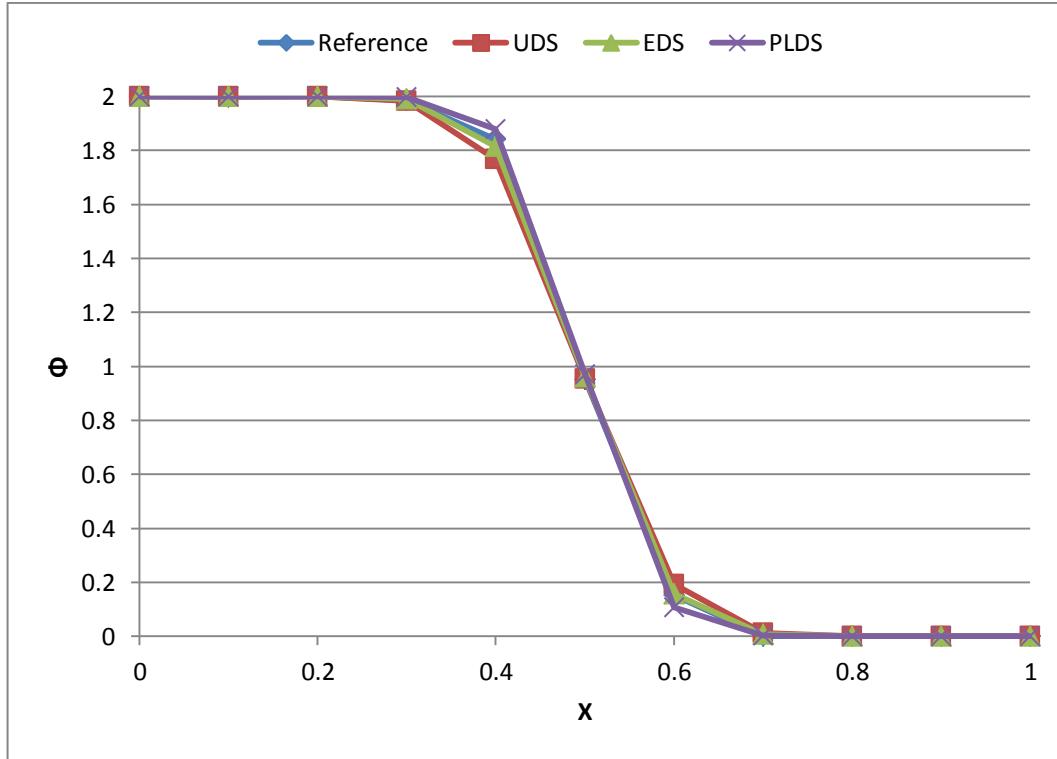


Figure 26: The Smith-Hutton Problem. Comparison between different numerical schemes  $\rho/\Gamma = 10^3$

#### 2.4.2.7. Conclusions

This problem has tested different low order numerical schemes for the computation of the solution of the convection-diffusion equation. During the analysis of the three different cases using the low order numerical schemes, some conclusions could be extracted.

First of all, for the case of  $\rho/\Gamma = 10$  the results do not vary significantly between the different numerical schemes and the committed absolute error with respect to the reference solution is almost equal. It has been observed that when the convective terms are not too much important with respect to the diffusive terms, with low densified meshes some good accuracy can be achieved for the right side of the outlet, but for the left side, a more densified mesh means more accuracy to the reference solution.

Analyzing the case of  $\rho/\Gamma = 10^3$ , where the convective terms are much more important than the diffusive terms, it is possible to find more differences between numerical schemes. First of all, the CDS scheme becomes instable due to the high Peclet number found in this case (this situation happens also with  $\rho/\Gamma = 10^6$ ). Then, it is seen that with more mesh density, more accurate results can be achieved, but some errors are observed with the PLDS scheme for high densified meshes, where the obtained values are beyond the limits of the reference solution (Figure 26). In this case, it is possible to affirm that the best numerical scheme to be used is the EDS scheme despite its extra computational time.

The last case to be analyzed is the case for  $\rho/\Gamma = 10^6$ , where the convective terms are really much more important than the diffusive terms. It can be seen that almost no diffusion occurs and that the variable is transported into the domain mainly due to the convective terms (*Figure 24*). At this point, no appreciable difference has been noticed between the different numerical schemes, since for high densified meshes the obtained results are almost equal using any of the studied numerical schemes.

Because of the abrupt boundary condition in the inlet, when the convective terms are stronger than the diffusive terms, a more refined mesh is needed in order to solve properly the values of the variable at the outlet. This fact can be seen in the *Figures 19, 21 and 23*. In *Figure 19*, with a non-densified mesh the obtained results are quite approximate to the reference solution while in *Figures 21 and 23* with a non-densified mesh the results are not good enough.

The effects of the diffusive term can be observed in *Figure 20*, where the variable is distributed along the domain. In *Figures 22 and 24* the convective terms are much more important and it can be seen as the variable flows according to the velocity field with almost no diffusion.

During the realization of this problem the limitations of the CDS scheme and the PLDS scheme could be noticed and studied. Then, the other two numerical schemes present more advantages. The main characteristics are that the UDS scheme is quicker than the other schemes, but the EDS scheme is much more accurate than the UDS scheme, although it takes more time to compute the results due to the exponential interpolation.

## 2.5. Introduction to the Fractional Step Method

Nowadays, one of the most used methods to solve the incompressible Navier-Stokes equations (dimensionless equations (83) and (84)) is the Fractional Step Method (FSM). Instead of using the normal resolution of the convective-diffusion equation and, despite its difficulties, the FSM has become popular due to its better performance and its code simplicity.

$$\frac{\partial \vec{u}}{\partial t} + (\vec{u} \cdot \nabla) \vec{u} = \frac{1}{Re} \Delta \vec{u} - \nabla p \quad (83)$$

$$\nabla \cdot \vec{u} = 0 \quad (84)$$

The idea behind the method is to project the velocity vector into a divergence-free velocity space; then, compute the predictor velocity in this space without any pressure gradient contribution and, finally, solve the Poisson equation (see equation (91)) in order to make the predictor velocity field incompressible and compute the actual velocity field.

This method uses the Helmholtz-Hodge theorem (see [3]), which states that a given vector field  $\vec{w}$ , defined in a bounded domain  $\Omega$  with smooth boundary  $\partial\Omega$ , is uniquely decomposed in a pure gradient field and a divergence-free vector parallel to  $\partial\Omega$ . Both spaces are orthogonal between themselves (see *Figure 27*). This theorem is expressed with the following equations:

$$\vec{w} = \vec{a} + \nabla\varphi \quad (85)$$

$$\nabla \cdot \vec{a} = 0, \quad \vec{a} \in \Omega \quad (86)$$

$$\vec{a} \cdot \vec{n} = 0, \quad \vec{a} \in \partial\Omega \quad (87)$$

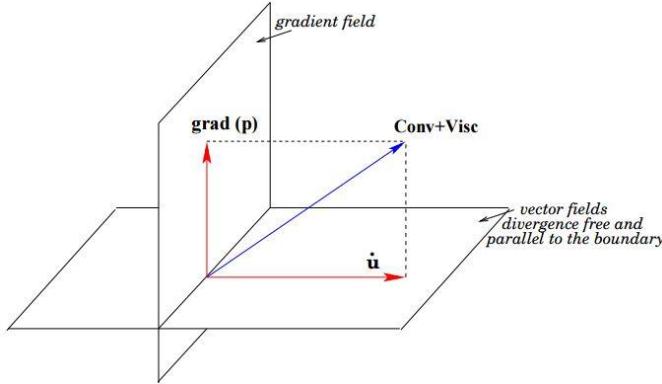


Figure 27: Convective + Viscous term vector field unique decomposition (extracted from [3])

Applying the Helmholtz-Hodge theorem to the incompressible Navier-Stokes equations using the projector operator  $\Pi$ , which projects any vector field onto a divergence-free space, the Poisson equation for pressure is found.

$$\Pi \left( \frac{\partial \vec{u}}{\partial t} + \nabla p \right) = \Pi \left( -(\vec{u} \cdot \nabla) \vec{u} + \frac{1}{Re} \Delta \vec{u} \right) \quad (88)$$

The velocity field is incompressible and, then, the transient term remains the same. Also the projection of the pressure gradient disappears.

$$\frac{\partial \vec{u}}{\partial t} = \Pi \left( -(\vec{u} \cdot \nabla) \vec{u} + \frac{1}{Re} \Delta \vec{u} \right) \quad (89)$$

Replacing the transient term into the incompressible Navier-Stokes equations and isolating the pressure gradient, it yields to:

$$\nabla p = -(\vec{u} \cdot \nabla) \vec{u} + \frac{1}{Re} \Delta \vec{u} - \Pi \left( -(\vec{u} \cdot \nabla) \vec{u} + \frac{1}{Re} \Delta \vec{u} \right) \quad (90)$$

If now the divergence operator is applied and the projector definition is used, the Poisson equation for pressure is found:

$$\Delta p = \nabla \cdot \left( -(\vec{u} \cdot \nabla) \vec{u} + \frac{1}{Re} \Delta \vec{u} \right) \quad (91)$$

### 2.5.1. Time-integration method

According to [3], a fully explicit time integration scheme is proposed for the sake of clarity when solving the FEM.

The first step is to simplify the notation. The momentum equation of Navier-Stokes equations can be rewritten (see equation (93)) using the notation  $R(\vec{u})$ .

$$R(\vec{u}) \equiv -(\vec{u} \cdot \nabla) \vec{u} + \frac{1}{Re} \Delta \vec{u} \quad (92)$$

$$\frac{\partial \vec{u}}{\partial t} = R(\vec{u}) - \nabla p \quad (93)$$

The semi-discretized Navier-Stokes equations can be obtained doing the following hypothesis: first of all, a CDS scheme is used when discretizing the time derivative term (see equation (94)). Then, in order to discretize the  $R(\vec{u})$  term, a fully explicit second-order Adams-Bashforth scheme is used (see equation (95)).

$$\left. \frac{\partial \vec{u}}{\partial t} \right|^{n+\frac{1}{2}} \approx \frac{\vec{u}^{n+1} - \vec{u}^n}{\Delta t} + O(\Delta t^2) \quad (94)$$

$$R^{n+\frac{1}{2}}(\vec{u}) \approx \frac{3}{2}R(\vec{u}^n) - \frac{1}{2}R(\vec{u}^{n-1}) + O(\Delta t^2, \Delta x^m) \quad (95)$$

To finish, a first-order backward Euler scheme for the pressure gradient and a fully implicit scheme for the incompressibility constraint are used. Then, the semi-discretized incompressible Navier-Stokes equations are written as follows:

$$\frac{\vec{u}^{n+1} - \vec{u}^n}{\Delta t} = \frac{3}{2}R(\vec{u}^n) - \frac{1}{2}R(\vec{u}^{n-1}) - \nabla p^{n+1} \quad (96)$$

$$\nabla \cdot \vec{u}^{n+1} = 0 \quad (97)$$

Using the Helmholtz-Hodge theorem explained before, it is possible to define a predictor velocity that can be decomposed into a divergence-free vector and into a gradient of a scalar field: the pseudo-pressure ( $\tilde{p} = \Delta t \cdot p^{n+1}$ ).

$$\vec{u}^p = \vec{u}^{n+1} + \nabla \tilde{p} \quad (98)$$

Replacing this equation into equation (96), it is rewritten as:

$$\vec{u}^p = \vec{u}^n + \Delta t \left( \frac{3}{2}R(\vec{u}^n) - \frac{1}{2}R(\vec{u}^{n-1}) \right) \quad (99)$$

Now, it is possible to solve the Navier-Stokes equations without taking into account the incompressibility constraint. After solving the predictor velocity, the incompressibility constraint can be forced applying the divergence operator to equation (98) in order to find again a Poisson equation for the pressure.

$$\nabla \cdot \vec{u}^p = \nabla \cdot \vec{u}^{n+1} + \nabla \cdot (\nabla \tilde{p}) \quad (100)$$

The equation (97) must be accomplished and it yields to the Poisson equation:

$$\Delta \tilde{p} = \nabla \cdot \vec{u}^p \quad (101)$$

When the Poisson equation is solved, it is possible to find the pressure scalar field that accomplishes equation (98), and the new incompressible velocity field (for the next time step) can be computed.

When solving the Fractional Step Method with a fully explicit scheme, it is necessary to pay special attention to the time step because of stability reasons. Therefore, according to [3], the following CFL conditions must be accomplished:

$$\Delta t \left( \frac{|u_i|}{\Delta x_i} \right)_{max} \leq C_{conv} \quad (102)$$

$$\Delta t \left( \frac{\nu}{\Delta x_i^2} \right)_{max} \leq C_{visc} \quad (103)$$

Where approximately  $C_{conv} = 0.35$  and  $C_{visc} = 0.2$  (see [4]).

## 2.6. Proposed problems: Fractional Step Method

### 2.6.1. The Driven Cavity

#### 2.6.1.1. Objective

This is a proposed problem by CTTC as an application of the Fractional Step Method. The objective of the problem is to solve the velocity field into a bi-dimensional cavity.

#### 2.6.1.2. Problem definition

The problem consists of a bi-dimensional square cavity whose top wall moves with a uniform velocity on its own plane and the other components of the velocity are restricted to zero on each wall. Furthermore, there exist also boundary conditions for the pressure field.

According to the upper wall velocity boundary condition, it is expected that the fluid will flow following a clockwise direction and two particularities will appear at the bottom corners of the cavity.

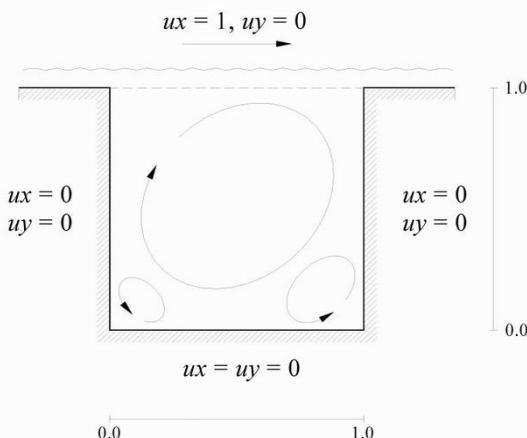


Figure 28: The Driven Cavity. Scheme of the problem (extracted from [5])

Then, the boundary conditions of the problem can be summarized as follows:

$$u = 1; v = 0; \frac{\partial p}{\partial y} = 0 \text{ (top cavity wall)} \quad (104)$$

$$u = 0; v = 0; \frac{\partial p}{\partial y} = 0 \text{ (bottom cavity wall)} \quad (105)$$

$$u = 0; v = 0; \frac{\partial p}{\partial x} = 0 \text{ (left cavity wall)} \quad (106)$$

$$u = 0; v = 0; \frac{\partial p}{\partial x} = 0 \text{ (right cavity wall)} \quad (107)$$

The cases of Reynolds numbers (equation (108)) 100, 400, 1000, 3200, 5000, 7500 and 10000 must be solved.

$$Re = \frac{\rho U x}{\mu} = \frac{U x}{\nu} \quad (108)$$

Where  $\rho$  is the density,  $\mu$  is the dynamic viscosity,  $\nu$  is the cinematic viscosity,  $U$  is a reference velocity and  $x$  is a reference distance.

#### 2.6.1.3. Numerical discretization

Before starting with the explanation of the numerical discretization of the problem, the used mesh must be explained. In this problem more than one variable must be computed: the horizontal component of the velocity field, the vertical component of the velocity field and the pressure. Since usually the velocity is computed at the faces of the cells, two staggered meshes have been used (one for each velocity component) additionally to the mesh shown in *Figure 6* (for the pressure field). Then, the staggered meshes are displaced half a control volume with respect to the pressure mesh.

The advantage of using more than one mesh is the simplicity when computing directly the velocity at the cell faces, which is needed for solving the Poisson equation. This advantage overcomes the disadvantages of having to deal with different meshes at the same time.

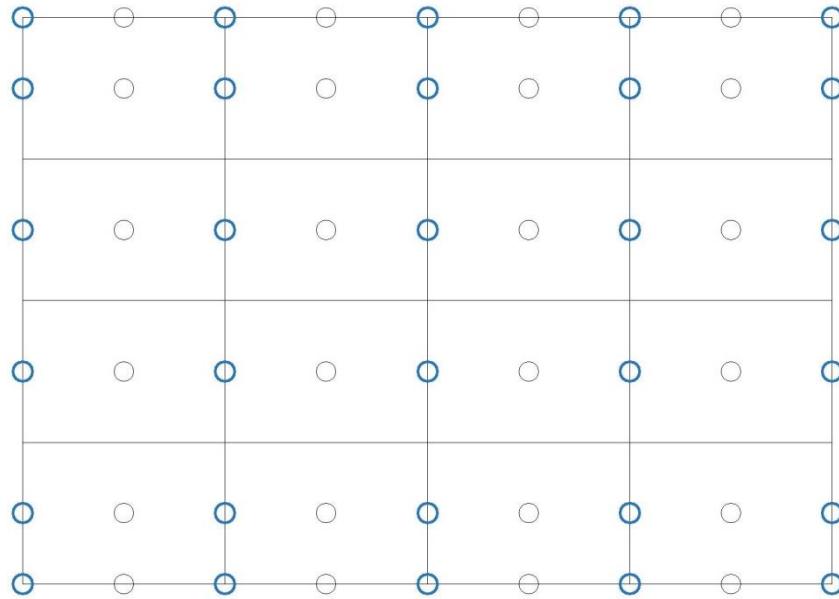


Figure 29: The Driven Cavity. U-component nodes (blue) superposed to the pressure mesh

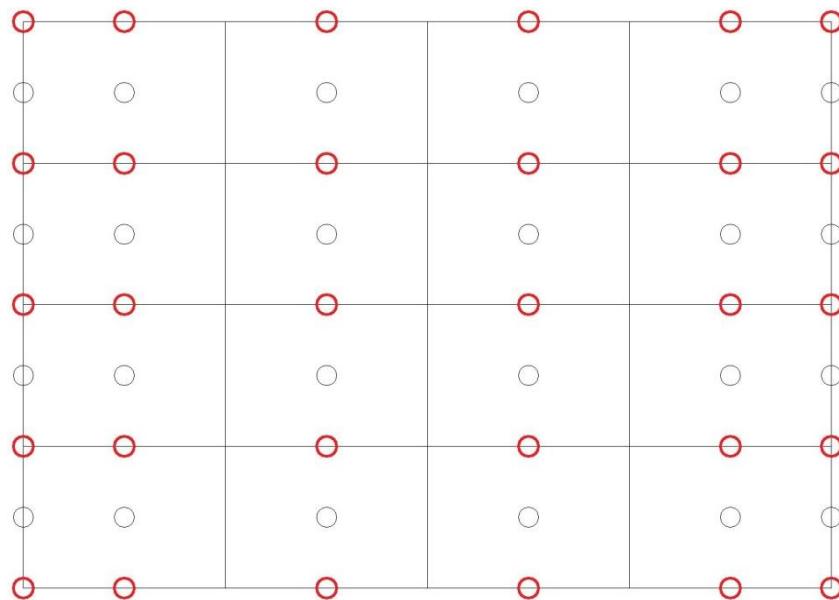


Figure 30: The Driven Cavity. V-component nodes (red) superposed to the pressure mesh

At this point, the equation (92) must be discretized. First of all, it can be seen that the operator R can be applied separately to each velocity component. Then,

$$R(u) = - \left( \frac{\partial u}{\partial x} u + \frac{\partial u}{\partial y} v \right) + \frac{1}{Re} \left( \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) \quad (109)$$

$$R(v) = - \left( \frac{\partial v}{\partial x} u + \frac{\partial v}{\partial y} v \right) + \frac{1}{Re} \left( \frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} \right) \quad (110)$$

Applying the same procedure with the Gauss Theorem seen in section 2.2, the previous equations, once discretized, are rewritten as:

$$R(u)\Delta x\Delta y = -u_e F_e + u_w F_w - u_n F_n + u_s F_s + \frac{\Delta y}{Re} \left( \frac{u_E - u_P}{\delta_{PE}} - \frac{u_P - u_W}{\delta_{PW}} \right) + \frac{\Delta x}{Re} \left( \frac{u_N - u_P}{\delta_{PN}} - \frac{u_P - u_S}{\delta_{PS}} \right) \quad (111)$$

$$R(v)\Delta x\Delta y = -v_e F_e + v_w F_w - v_n F_n + v_s F_s + \frac{\Delta y}{Re} \left( \frac{v_E - v_P}{\delta_{PE}} - \frac{v_P - v_W}{\delta_{PW}} \right) + \frac{\Delta x}{Re} \left( \frac{v_N - v_P}{\delta_{PN}} - \frac{v_P - v_S}{\delta_{PS}} \right) \quad (112)$$

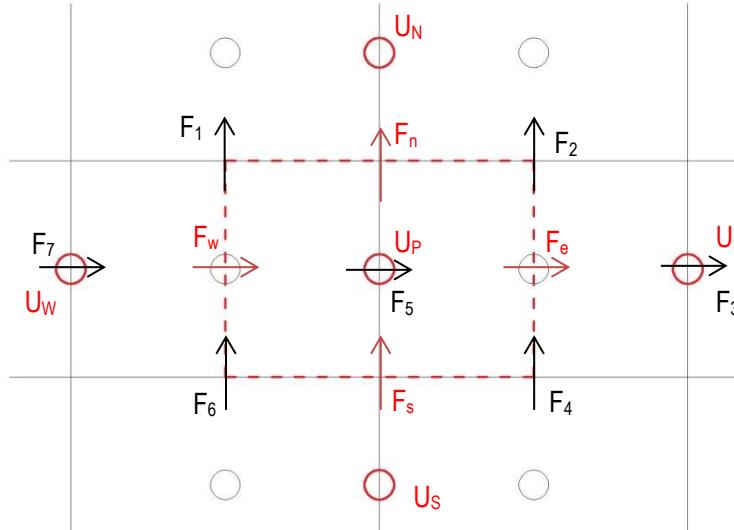


Figure 31: The Driven Cavity. Schema of the displaced control volume

It is important to comment how to compute or estimate the different variables that appear in equation (111) and equation (112). The general schema of the distribution of the variables around a generic node P of the U-component mesh is shown in *Figure 31*. The explanation that follows can be also applied to a generic node P of the V-component mesh.

As it can be seen in *Figure 31*, the node P of the U-component mesh has a cell associated to it. Then, the main question that arises is how to compute the U-component ( $u_i$ ) and the fluxes  $F_i$  at the faces of the cell.

The velocity component at the cell faces can be computed using any of the low order numerical schemes. Since the formulation is fully explicit in this case, a CDS scheme has been used to compute the values of the velocity components. Additionally, the fluxes have been calculated using also an average between the known fluxes. Since the used mesh has equal control volumes, the formulation is as simple as a CDS scheme. If the cells were not equal, some ponderation should be done:

$$F_n = \frac{1}{2} (F_1 + F_2) \quad (113)$$

$$F_s = \frac{1}{2} (F_4 + F_6) \quad (114)$$

$$F_e = \frac{1}{2} (F_3 + F_5) \quad (115)$$

$$F_w = \frac{1}{2} (F_5 + F_7) \quad (116)$$

Note that with this scheme the conservation of mass is still accomplished. Assuming that all the cells are equal and quadrilateral ( $\Delta x = \Delta y$ ), the mass conservation can be expressed as a function of the fluxes of the two control volumes of the pressure mesh used to compute the fluxes of the cell of the staggered mesh:

$$F_e - F_w + F_n - F_s = \frac{1}{2} (F_1 - F_6 + F_5 - F_7) + \frac{1}{2} (F_2 - F_4 + F_3 - F_5) = 0 \quad (117)$$

The other equation that must be discretized is the Poisson equation (101). In fact, this equation is the one that must be solved with the developed solver for each cell. Once discretized using the Gauss Theorem, it is rewritten as:

$$\left( \left( \frac{\partial p}{\partial x} \right)_e - \left( \frac{\partial p}{\partial x} \right)_w \right) \Delta y + \left( \left( \frac{\partial p}{\partial y} \right)_n - \left( \frac{\partial p}{\partial y} \right)_s \right) \Delta x \\ = (u_e^p - u_w^p) \Delta y + (v_n^p - v_s^p) \Delta x \quad (118a)$$

$$\frac{p_E - p_P}{\delta_{PE}} \Delta y - \frac{p_P - p_W}{\delta_{PW}} \Delta y + \frac{p_N - p_P}{\delta_{PN}} \Delta x - \frac{p_P - p_S}{\delta_{PS}} \Delta x \\ = (u_e^p - u_w^p) \Delta y + (v_n^p - v_s^p) \Delta x \quad (118b)$$

The discretization coefficients for the Poisson equation are shown in the following tables.

Top cavity wall	Value	Bottom cavity wall	Value	Upper-left corner	Value	Upper-right corner	Value
$a_P$	1	$a_P$	1	$a_P$	1	$a_P$	1
$a_E$	0	$a_E$	0	$a_E$	0.5	$a_E$	0
$a_W$	0	$a_W$	0	$a_W$	0	$a_W$	0.5
$a_N$	0	$a_N$	1	$a_N$	0	$a_N$	0
$a_S$	1	$a_S$	0	$a_S$	0.5	$a_S$	0.5
$b_P$	0	$b_P$	0	$b_P$	0	$b_P$	0
Left cavity wall	Value	Right cavity wall	Value	Lower-left corner	Value	Lower-right corner	Value
$a_P$	1	$a_P$	1	$a_P$	1	$a_P$	1
$a_E$	1	$a_E$	0	$a_E$	0.5	$a_E$	0
$a_W$	0	$a_W$	1	$a_W$	0	$a_W$	0.5
$a_N$	0	$a_N$	0	$a_N$	0.5	$a_N$	0.5
$a_S$	0	$a_S$	0	$a_S$	0	$a_S$	0
$b_P$	0	$b_P$	0	$b_P$	0	$b_P$	0

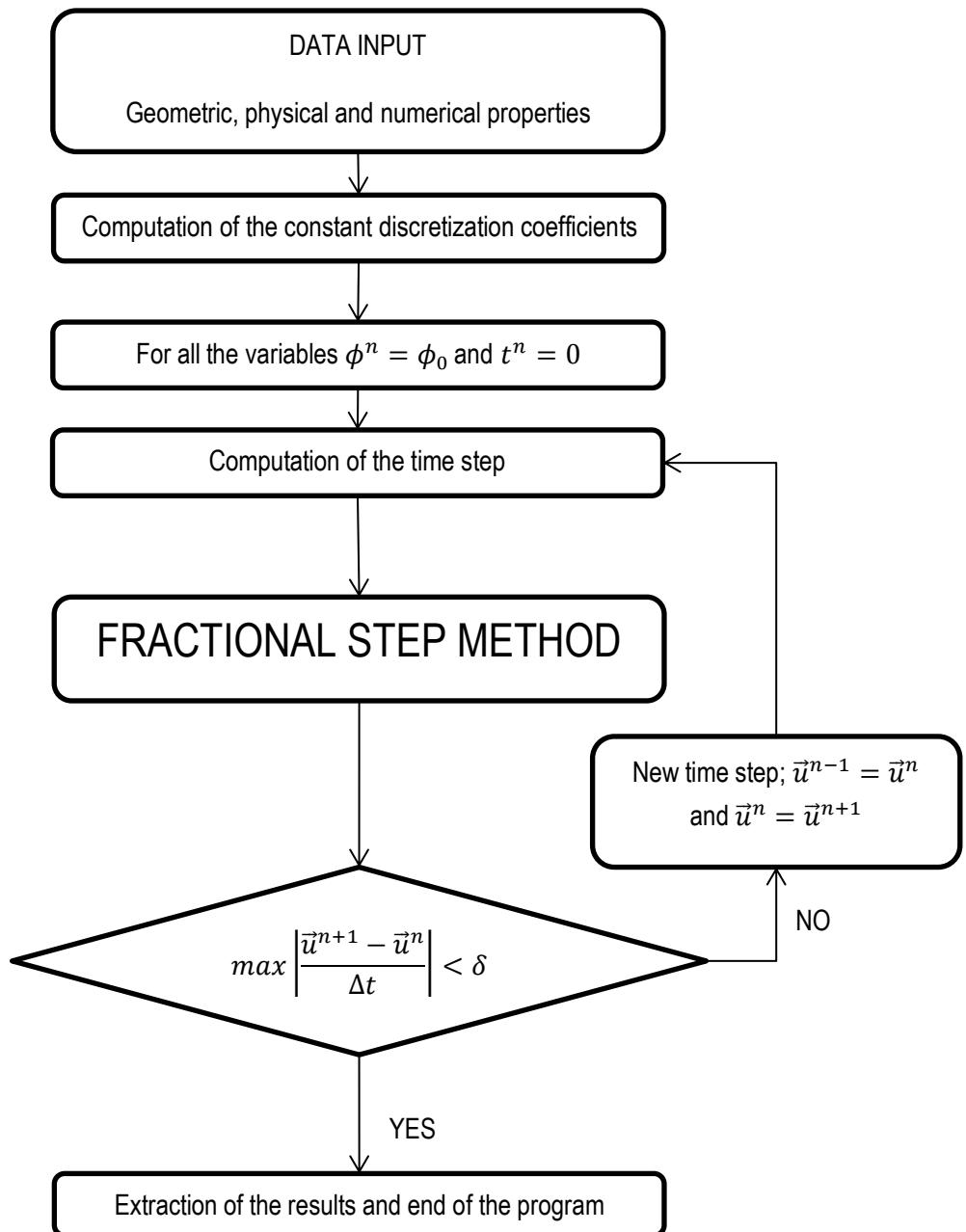
Table 14: The Driven Cavity. Discretization coefficients of boundary nodes of the pressure field

Inner nodes	Value
$a_E$	$\frac{\Delta y}{\delta_{PE}}$
$a_W$	$\frac{\Delta y}{\delta_{PW}}$
$a_N$	$\frac{\Delta x}{\delta_{PN}}$
$a_S$	$\frac{\Delta x}{\delta_{PS}}$
$a_P$	$a_N + a_S + a_E + a_W$
$b_P$	$-(u_e^p - u_w^p)\Delta y - (v_n^p - v_s^p)\Delta x$

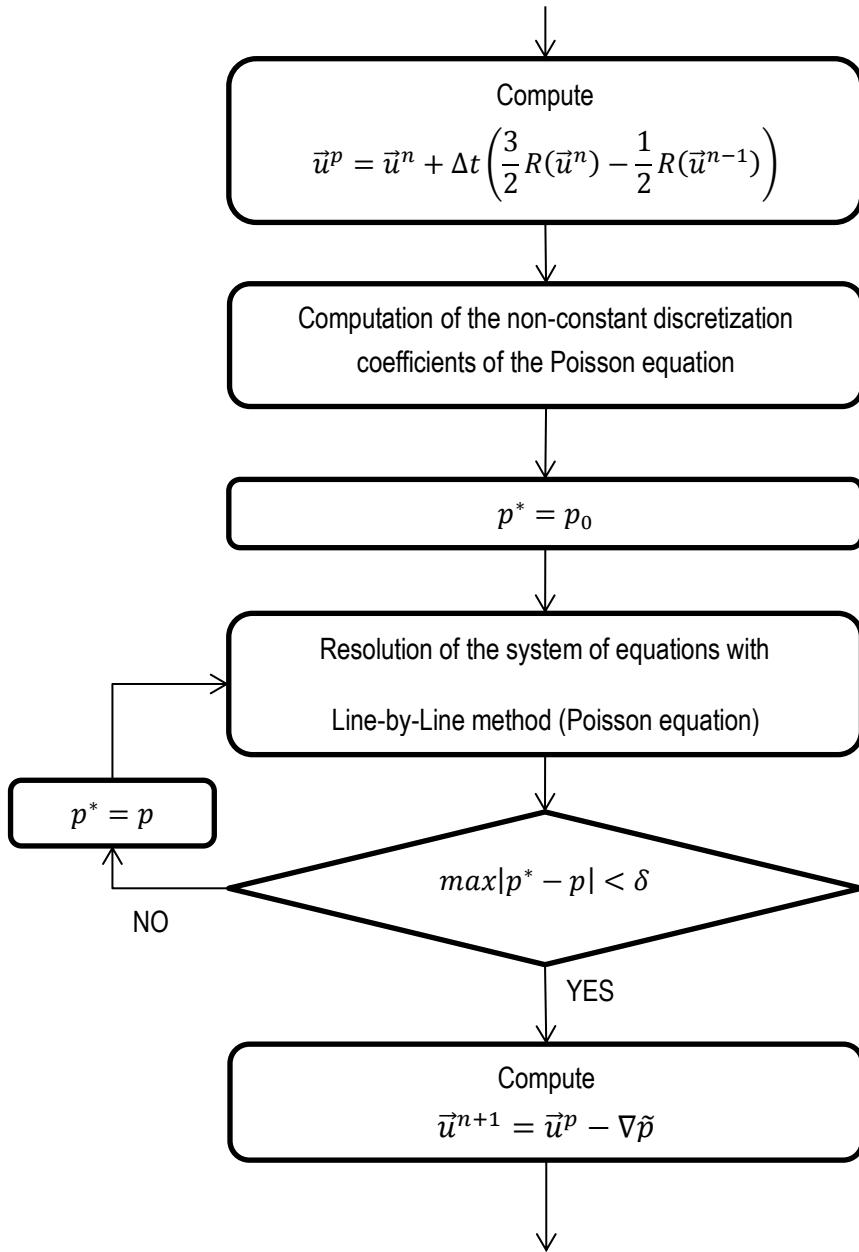
Table 15: The Driven Cavity. Discretization coefficients of inner nodes of the pressure field

One point that must be clarified refers to the boundary coefficients. Since all the boundary conditions are from Neumann type, one node must have a fixed value (Dirichlet type) in order to be able to solve the system of equations. According to that, it has been chosen the node in the middle of the bottom cavity wall to have a fixed value of zero. For doing that, this node has  $a_P = 1$  and the rest of the discretization coefficients equal to zero.

#### 2.6.1.4. Global algorithm of resolution



FRACTIONAL STEP METHOD black box:



#### 2.6.1.5. Results

In this section, the most relevant results for the Driven Cavity problem are shown.

The reference solution is shown in the following tables (see [6]), where the values for the U-component of the velocity are extracted in the middle vertical line of the cavity and the values for the V-component are extracted in the middle horizontal line of the cavity.

U-component	Reynolds Number (Re)						
	100	400	1000	3200	5000	7500	10000
y							
1	1	1	1	1	1	1	1
0.9766	0.84123	0.75837	0.65928	0.53236	0.48223	0.47244	0.47221
0.9688	0.78871	0.68439	0.57492	0.48296	0.4612	0.47048	0.47783
0.9609	0.73722	0.61756	0.51117	0.46547	0.45992	0.47323	0.4807
0.9531	0.68717	0.55892	0.46604	0.46101	0.46036	0.47167	0.47804
0.8516	0.23151	0.29093	0.33304	0.34682	0.33556	0.34228	0.34635
0.7344	0.00332	0.16256	0.18719	0.19791	0.20087	0.20591	0.20673
0.6172	-0.13641	0.02135	0.05702	0.07156	0.08183	0.08342	0.08344
0.5	-0.20581	-0.11477	-0.0608	-0.04272	-0.03039	-0.038	-0.03111
0.4531	-0.2109	-0.17119	-0.10648	-0.86636	-0.07404	-0.07503	-0.0754
0.2813	-0.15662	-0.32726	-0.27805	-0.24427	-0.22855	-0.23176	-0.23186
0.1719	-0.1015	-0.24299	-0.38289	-0.34323	-0.3305	-0.32393	-0.32709
0.1016	-0.06434	-0.14612	-0.2973	-0.41932	-0.40435	-0.38324	-0.38
0.0703	-0.04775	-0.10338	-0.2222	-0.37827	-0.43643	-0.43025	-0.41657
0.0625	-0.04192	-0.09266	-0.20196	-0.35344	-0.42901	-0.4359	-0.42537
0.0547	-0.03717	-0.08186	-0.18109	-0.32407	-0.41165	-0.43154	-0.42735
0	0	0	0	0	0	0	0

Table 16: The Driven Cavity. Reference solution (U-component)

V-component	Reynolds Number (Re)						
	100	400	1000	3200	5000	7500	10000
x							
1	0	0	0	0	0	0	0
0.9688	-0.05906	-0.12146	-0.21388	-0.39017	-0.49774	-0.53858	-0.54302
0.9609	-0.07391	-0.15663	-0.27669	-0.47425	-0.55069	-0.55216	-0.52987
0.9531	-0.08864	-0.19254	-0.33714	-0.52357	-0.55408	-0.52347	-0.49099
0.9453	-0.10313	-0.22847	-0.39188	-0.54053	-0.52876	-0.4889	-0.45863
0.9063	-0.16914	-0.23827	-0.5155	-0.44307	-0.41442	-0.4105	-0.41496
0.8594	-0.22445	-0.44993	-0.42665	-0.37401	-0.36214	-0.36213	-0.36737
0.8047	-0.24533	-0.38598	-0.31966	-0.31184	-0.30018	-0.30448	-0.30719
0.5	0.05454	0.05186	0.02526	0.00999	0.00945	0.00824	0.00831
0.2344	0.17527	0.30174	0.32235	0.28188	0.2728	0.27348	0.27224
0.2266	0.17507	0.30203	0.33075	0.2903	0.28066	0.28117	0.28003
0.1563	0.16077	0.28124	0.37095	0.37119	0.35368	0.3506	0.3507
0.0938	0.12317	0.22965	0.32627	0.42768	0.42951	0.41824	0.41487
0.0781	0.1089	0.2092	0.30353	0.41906	0.43648	0.43564	0.43124
0.0703	0.10091	0.19713	0.29012	0.40917	0.43329	0.4403	0.43733
0.0625	0.09233	0.1836	0.27485	0.3956	0.42447	0.43979	0.43983
0	0	0	0	0	0	0	0

Table 17: The Driven Cavity. Reference solution (V-component)

The following figures show the results of the velocity components obtained for the Reynolds numbers 100, 5000 and 10000 with the mesh refinements 50x50, 80x80 and 125x125 nodes.

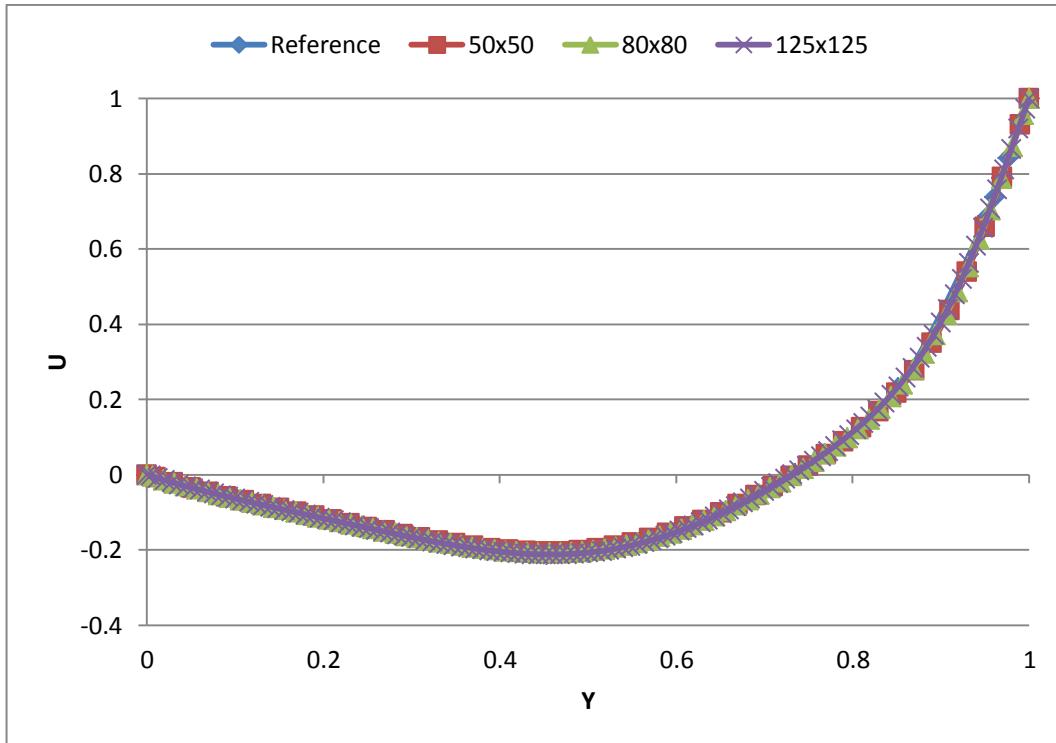


Figure 32: The Driven Cavity. Obtained results for U velocity component in the vertical mid-plane with different mesh sizes and  $Re = 100$

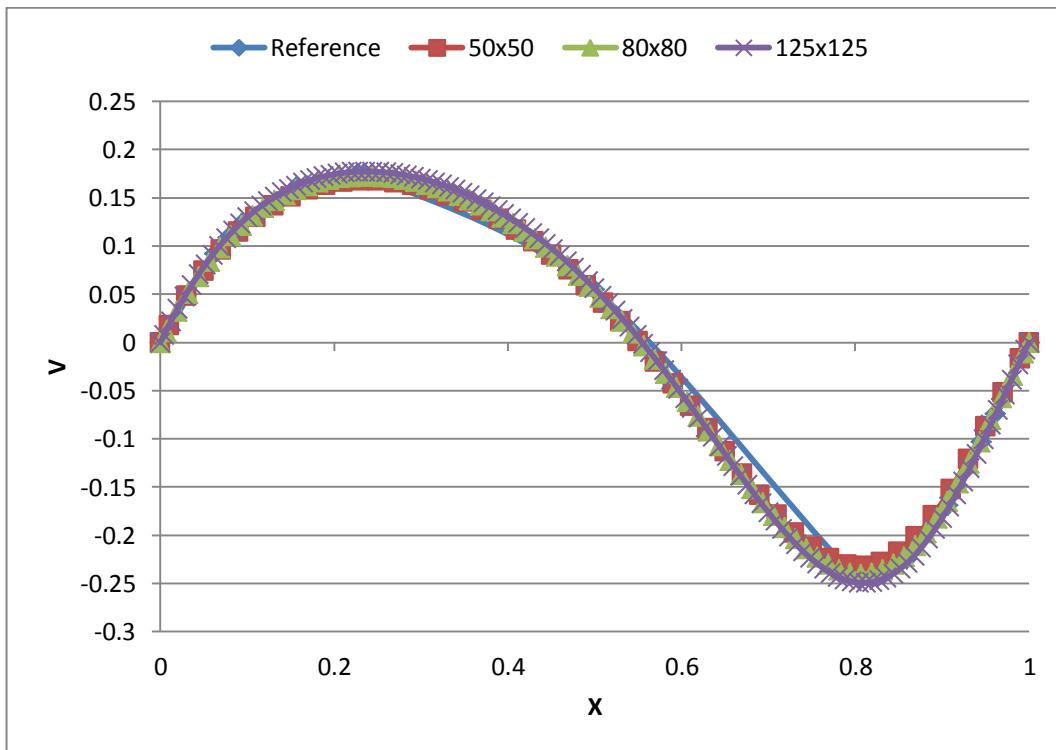


Figure 33: The Driven Cavity. Obtained results for V velocity component in the horizontal mid-plane with different mesh sizes and  $Re = 100$

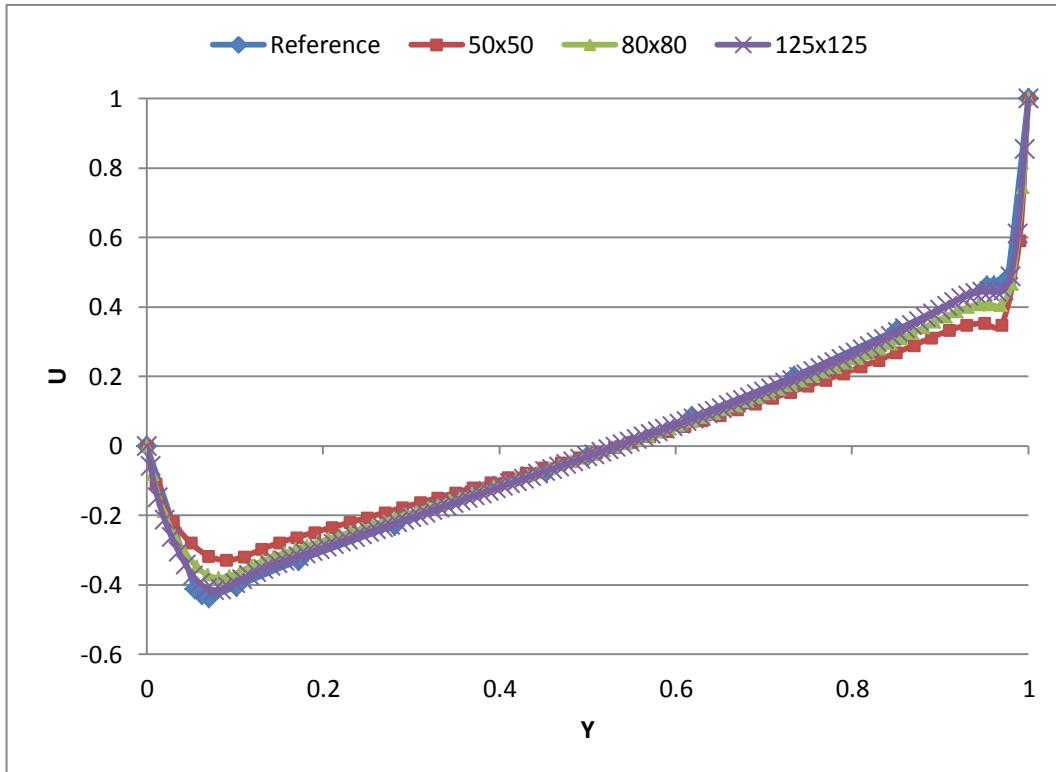


Figure 34: The Driven Cavity. Obtained results for U velocity component in the vertical mid-plane with different mesh sizes and  $Re = 5000$

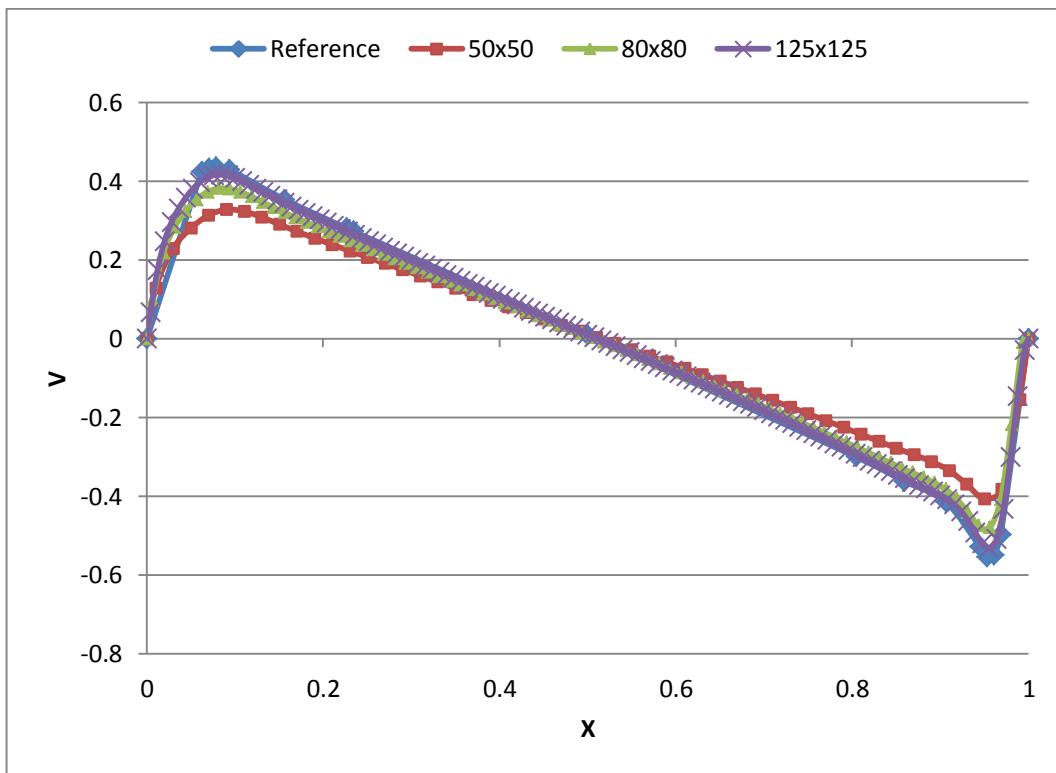


Figure 35: The Driven Cavity. Obtained results for V velocity component in the horizontal mid-plane with different mesh sizes and  $Re = 5000$

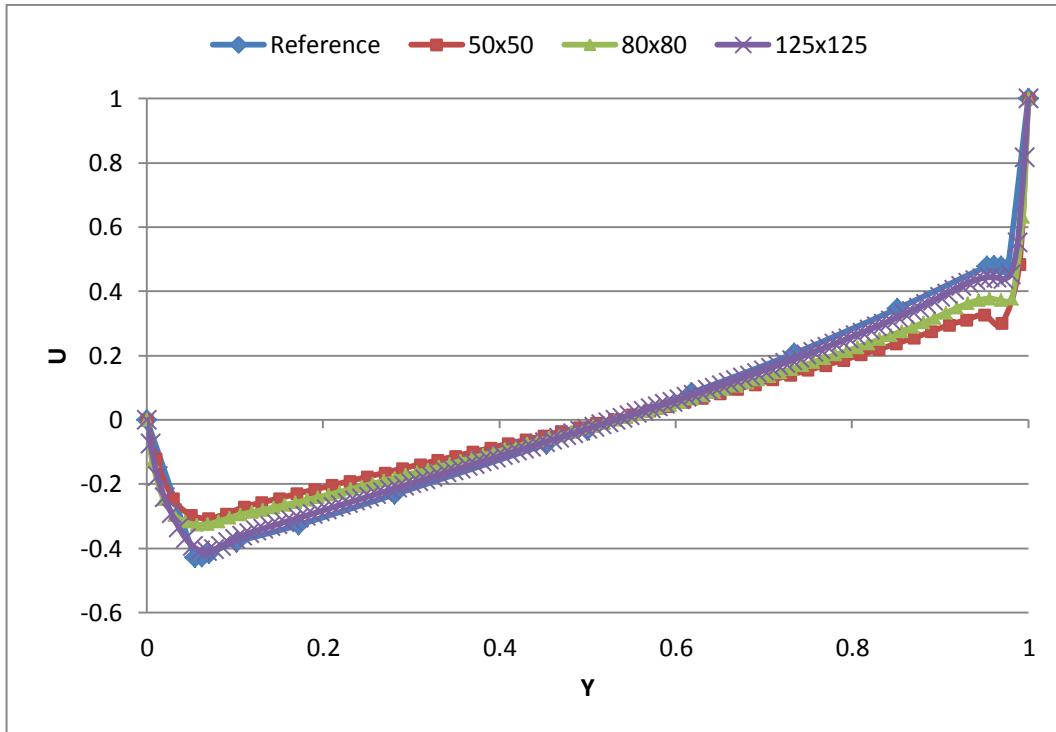


Figure 36: The Driven Cavity. Obtained results for U velocity component in the vertical mid-plane with different mesh sizes and  $Re = 10000$

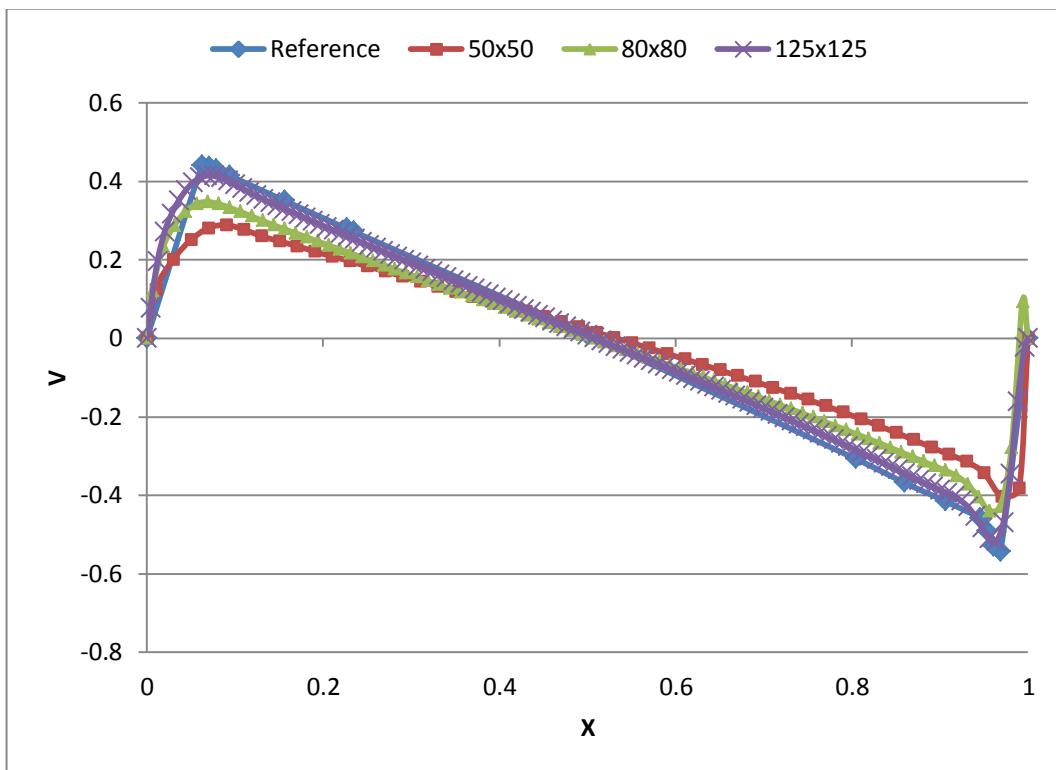
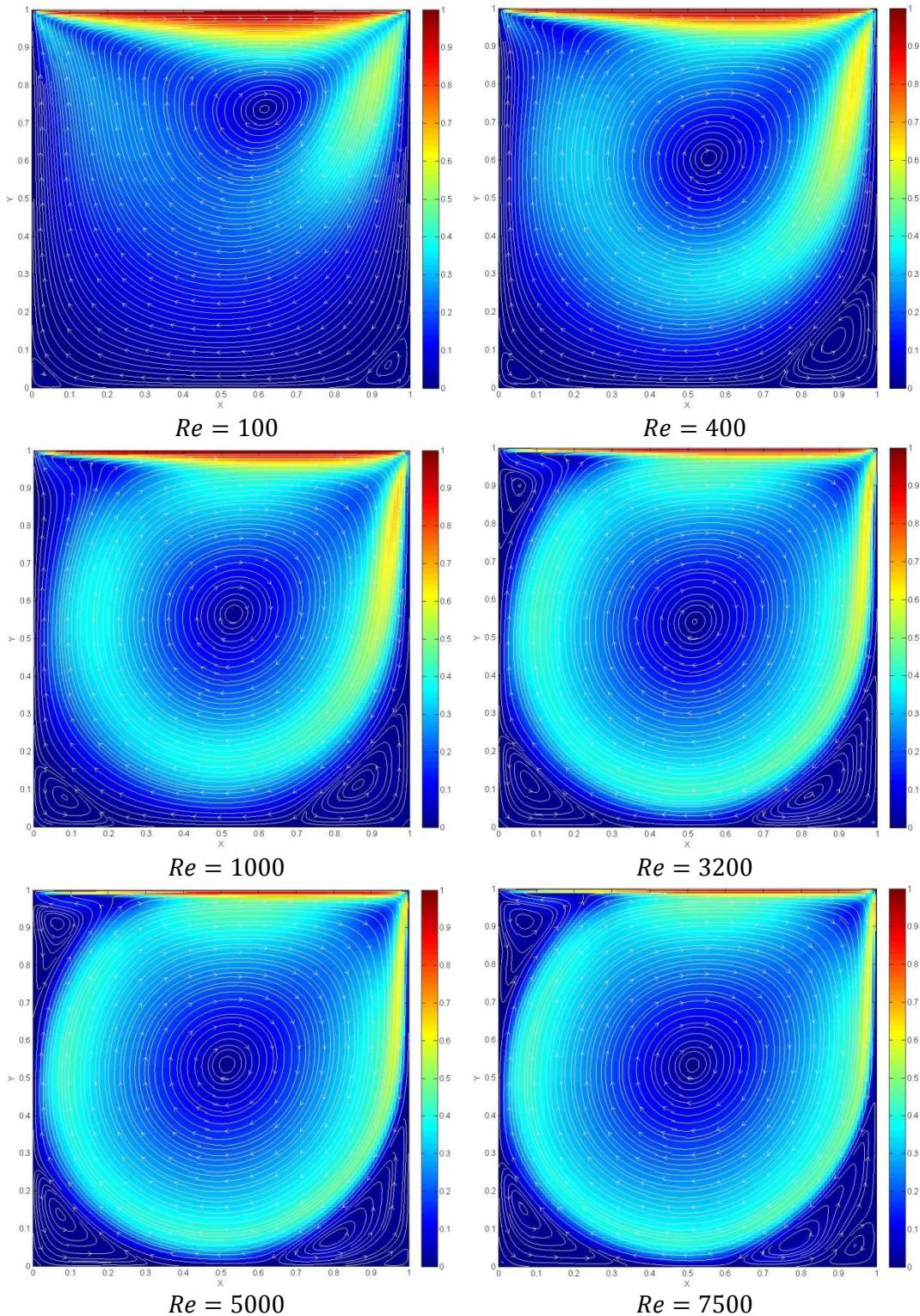


Figure 37: The Driven Cavity. Obtained results for V velocity component in the horizontal mid-plane with different mesh sizes and  $Re = 10000$



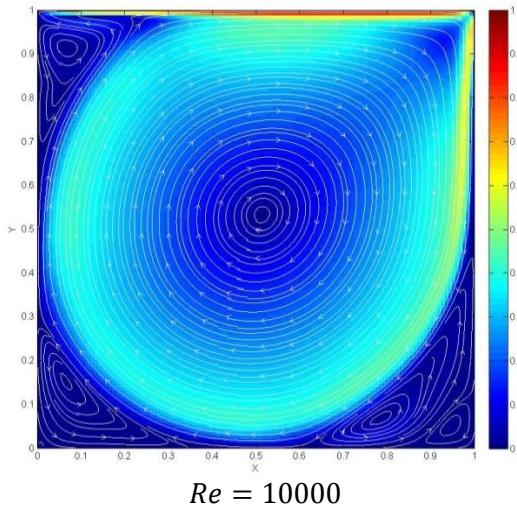


Figure 38: The Driven Cavity. Velocity module and streamlines for different Reynolds numbers

#### 2.6.1.6. Conclusions

This problem has computed the solution of the velocity field inside a quadrangular cavity whose top wall moves with a uniform velocity  $u = 1$  and  $v = 0$ . In order to solve the Navier-Stokes equations, the Fractional Step Method proposed in [3] has been used. Then, some different cases with different Reynolds numbers have been solved and some conclusions could be extracted.

For all Reynolds numbers, between  $100 \leq Re \leq 10000$ , a steady solution could be achieved for every tested mesh (50x50, 80x80 and 125x125 nodes) with a convergence criteria of  $\varepsilon < 10^{-7}$ , except for the case of  $Re = 10000$  and a mesh of 80x80 nodes, where the simulation was done until 1600 seconds. Also, another parameter that influences the convergence of the solution is the chosen time step for each iteration. Due to the low refinement of the mesh, the convective terms usually determine the time step. As seen in [3], a common value for  $C_{conv} = 0.35$  is taken, but in some cases ( $Re = 1000$  and  $Re = 3200$ ) this value should be reduced to  $C_{conv} = 0.1$  (it means having a lower time step) in order to have a convergent solution.

As the Reynolds number increases, it can be seen as the reference solution can be only achieved if the mesh is more densified. For example, for  $Re = 100$ , with a mesh of 50x50 nodes, the obtained solution is almost equal to the reference solution and with a mesh of 80x80 nodes, no difference can be appreciated. However, for  $Re = 5000$ , only with the mesh of 125x125 nodes a solution close to the reference solution can be obtained. For the last studied case ( $Re = 10000$ ), even the 125x125 mesh is not enough and more mesh refinement should be done. This phenomenon happens because for high Reynolds numbers, the flux starts to become turbulent and the small scales of the turbulent flow cannot be captured with the used meshes.

For all the tested Reynolds numbers, it appears a main vortex in the middle of the cavity following a clockwise direction (since the top cavity wall moves towards the right, the direction of the main vortex makes sense). Increasing the Reynolds number, three more vortexes start to appear. Then, for relative high Reynolds numbers (from  $Re = 3200$ ), three vortexes can be recognized in the

upper-left corner, lower-left corner and lower-right corner. Also the direction of these vortexes can be extracted from the *Figure 38*. The vortexes in the lower-left and in the lower-right corner follow a non-clockwise direction (see also *Figure 28*). These previous vortexes have a logical appearance, since the main vortex moves the fluid around them, causing their non-clockwise motions, but the third vortex (upper-left corner) moves also in a non-clockwise direction when it has two opposite circulatory movements around it. The main vortex moves the fluid in order to make the vortex follow a non-clockwise direction, while the upper boundary condition moves the fluid in order to cause a clockwise movement. It could be possible that the main vortex has a stronger influence to that third vortex than the boundary condition. However, this third vortex should be considered as a singularity of the problem. Also for  $Re = 7500$  and  $Re = 10000$  a fourth small vortex appears in the lower-right corner, as it does in the reference solution (see [6]).

### 2.6.2. Differentially Heated Cavity

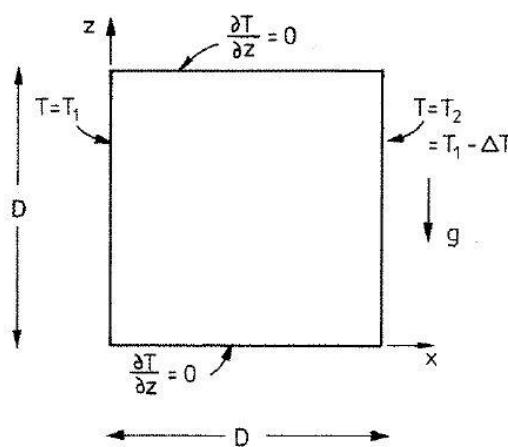
#### 2.6.2.1. Objective

This is a proposed problem by CTTC as an application of the Fractional Step Method. The objective of the problem is to solve the velocity field into a bi-dimensional cavity with natural convection caused by a temperature difference between the left and the right wall of the cavity. All the information regarding this case can be found in [7] and [8].

#### 2.6.2.2. Problem definition

The problem consists in a bi-dimensional square cavity which has enclosed a Boussinesq fluid of Prandtl number 0.71 inside, so the velocity components are imposed to be zero on each wall with a temperature difference between the left and the right wall of the cavity. Furthermore, boundary conditions for the pressure field must be imposed.

The following figure shows the schema of the problem:



**Figure 39: Differentially Heated Cavity. Scheme of the problem (extracted from [8])**

Then, the boundary conditions of the problem can be summarized as follows:

$$\vec{v} = \vec{0}; \frac{\partial p}{\partial z} = 0; \frac{\partial T}{\partial z} = 0 \text{ (top cavity wall)} \quad (119)$$

$$\vec{v} = \vec{0}; \frac{\partial p}{\partial z} = 0; \frac{\partial T}{\partial z} = 0 \text{ (bottom cavity wall)} \quad (120)$$

$$\vec{v} = \vec{0}; \frac{\partial p}{\partial x} = 0; T = 1 \text{ (left cavity wall)} \quad (121)$$

$$\vec{v} = \vec{0}; \frac{\partial p}{\partial x} = 0; T = 0 \text{ (right cavity wall)} \quad (122)$$

The flow and the thermal field must be computed for the Rayleigh numbers  $10^3$ ,  $10^4$ ,  $10^5$  and  $10^6$ .

The Prandtl number (equation (123)) and the Rayleigh number (equation (124)) are defined as follows:

$$Pr = \frac{\nu}{\kappa} = \frac{1}{Re} \quad (123)$$

$$Ra = \frac{g\beta\Delta TD^3}{\kappa\nu} \quad (124)$$

#### 2.6.2.3. Numerical discretization

This problem is very similar to the Driven Cavity seen before, but the contribution of the natural convection has to be included assuming a Boussinesq fluid. Then, the governing equations to be solved are the incompressible Navier-Stokes equations (125) and (126) and a convection-diffusion equation for the temperature (127):

$$\nabla \cdot \vec{v} = 0 \quad (125)$$

$$\rho \frac{\partial \vec{v}}{\partial t} + \rho(\vec{v} \cdot \nabla)\vec{v} = -\nabla p + \mu \nabla^2 \vec{v} + \rho \vec{g} \beta (T - T_\infty) \quad (126)$$

$$\frac{\partial T}{\partial t} + \vec{v} \cdot \nabla T = \kappa \nabla^2 T \quad (127)$$

Expressing them in non-dimensional variables, it yields to:

$$\nabla \cdot \vec{v} = 0 \quad (128)$$

$$\frac{\partial \vec{v}}{\partial t} + (\vec{v} \cdot \nabla)\vec{v} = -\nabla p + Pr \nabla^2 \vec{v} + Ra Pr T \quad (129)$$

$$\frac{\partial T}{\partial t} + \vec{v} \cdot \nabla T = \nabla^2 T \quad (130)$$

Here  $\vec{v}$ ,  $t$  and  $T$  are the non-dimensional velocity, time and temperature.

As said before, this problem shares many common things with the Driven Cavity problem; hence the domain discretization and the used mesh are the same. Pressure and temperature share the type of mesh shown in *Figure 6* while the velocity components still have the meshes shown in *Figures 29 and 30*.

This problem combines the Fractional Step Method to solve the velocity field with the normal solver designed for the resolution of the convection-diffusion equation. The main part of the code to be changed is when computing equation (92) in the vertical direction of the domain (since gravity points toward this direction). The discretized version of the vertical component of equation (92) is shown in equation (111), but now the contribution of the natural convection seen in equation (129) must be included as follows:

$$R(v)\Delta x \Delta y = -v_e F_e + v_w F_w - v_n F_n + v_s F_s + \frac{\Delta y}{Re} \left( \frac{v_E - v_P}{\delta_{PE}} - \frac{v_P - v_W}{\delta_{PW}} \right) + \frac{\Delta x}{Re} \left( \frac{v_N - v_P}{\delta_{PN}} - \frac{v_P - v_S}{\delta_{PS}} \right) + Ra Pr T \Delta x \Delta y \quad (131)$$

Another important point of this problem is the computation of the Nusselt number, since it is one of the parameters used to validate the results. This number expresses the ratio of convective to conductive heat transfer normal to a boundary and can be understood as the non-dimensional form of the heat flux. Then, the local Nusselt number can be defined as:

$$\overrightarrow{Nu}(x, z) = \vec{v}T - \nabla T \quad (132)$$

Where  $\vec{v}$  and  $T$  are the non-dimensional velocity and temperature.

The validation results only ask for the Nusselt number in the x-direction, so the previous equation is expressed as follows:

$$Nu_x = uT - \frac{\partial T}{\partial x} \quad (133)$$

Another result to be given is the average Nusselt number (equation (134)) at the walls of the cavity.

$$\overline{Nu}_x = \int_0^1 Nu_x(x, z) dz \quad (134)$$

There have been implemented two different ways to compute it: a simple discrete integration (equation (135)) and a Simpson's rule [9] (equation (136)).

$$\overline{Nu}_x = \frac{1}{z_f - z_i} \sum_k Nu_x(x, z_k) \cdot \Delta z_k \quad (135)$$

$$\overline{Nu}_x = \frac{z_f - z_i}{6} \left( Nu_x(x, z_i) + 4Nu_x \left( x, \frac{z_i + z_f}{2} \right) + Nu_x(x, z_f) \right) \quad (136)$$

About the resolution of the Poisson equation, there is no difference with respect to what is explained in the Driven Cavity problem and the coefficients of the system of equations are the ones that appear in the Tables 14 and 15.

When writing the system of equations to solve the temperature field, the following coefficients have been taken for the boundary nodes:

Top cavity wall	Value	Bottom cavity wall	Value	Upper-left corner	Value	Upper-right corner	Value
$a_P$	1	$a_P$	1	$a_P$	1	$a_P$	1
$a_E$	0	$a_E$	0	$a_E$	0.5	$a_E$	0
$a_W$	0	$a_W$	0	$a_W$	0	$a_W$	0.5
$a_N$	0	$a_N$	1	$a_N$	0	$a_N$	0
$a_S$	1	$a_S$	0	$a_S$	0.5	$a_S$	0.5
$b_P$	0	$b_P$	0	$b_P$	0	$b_P$	0
Left cavity wall	Value	Right cavity wall	Value	Lower-left corner	Value	Lower-right corner	Value
$a_P$	1	$a_P$	1	$a_P$	1	$a_P$	1
$a_E$	0	$a_E$	0	$a_E$	0.5	$a_E$	0
$a_W$	0	$a_W$	0	$a_W$	0	$a_W$	0.5
$a_N$	0	$a_N$	0	$a_N$	0.5	$a_N$	0.5
$a_S$	0	$a_S$	0	$a_S$	0	$a_S$	0
$b_P$	1	$b_P$	0	$b_P$	0	$b_P$	0

Table 18: Differentially Heated Cavity. Discretization coefficients of boundary nodes of the temperature field

The coefficients for the inner nodes follow the equations (25) to (32) with a CDS numerical scheme.

Note that in this problem the component  $v$  of the velocity corresponds to the axis direction  $z$  of Figure 39.

The global algorithm of resolution is the same as in the Driven Cavity problem, but adding a solver for the temperature field before starting another time step.

#### 2.6.2.4. Results

According to [8], the following results should be supplied:

- Flow and thermal field.
- Average Nusselt number.
- Maximum and minimum local Nusselt numbers on the hot wall and their location.
- Maximum vertical velocity on the horizontal mid-plane and its location.
- Maximum horizontal velocity on the vertical mid-plane and its location.
- Contour plots of the velocity components, the stream function, pressure and vorticity.

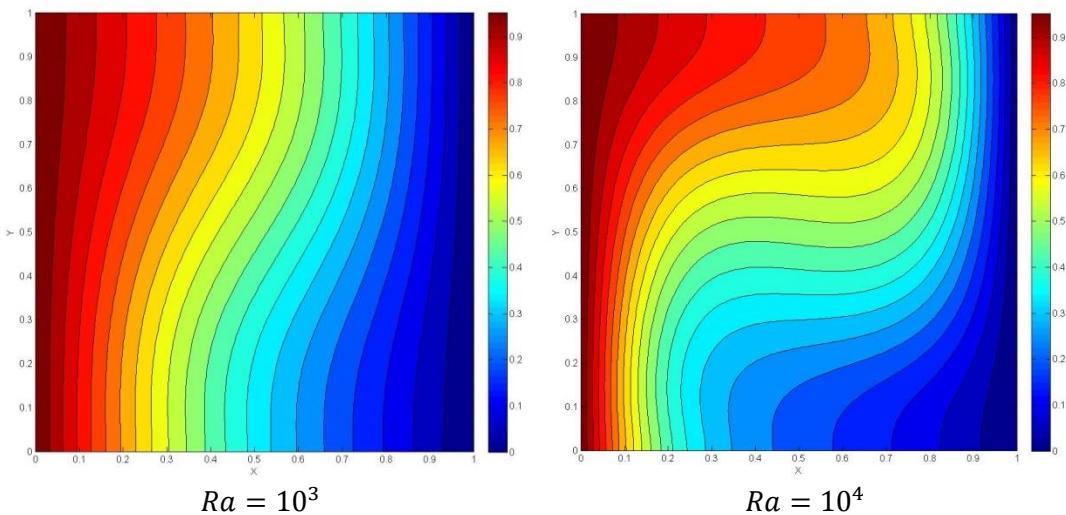
	Ra			
	<b>10<sup>3</sup></b>	<b>10<sup>4</sup></b>	<b>10<sup>5</sup></b>	<b>10<sup>6</sup></b>
$u_{Max}$	3.649	16.178	34.73	64.63
$z$	0.813	0.823	0.855	0.850
$v_{Max}$	3.697	19.617	68.59	219.36
$x$	0.178	0.119	0.066	0.0379
$\bar{Nu}$	1.118	2.243	4.519	8.800
$Nu_{Max}$	1.505	3.528	7.717	17.925
$z$	0.092	0.143	0.081	0.0378
$Nu_{Min}$	0.692	0.586	0.729	0.989
$z$	1	1	1	1

Table 19: Differentially Heated Cavity. Reference results of the problem (see [8])

The reference solution for the contour plots can be found in [7]. Here the most relevant results are presented. The results have been computed for 10x10, 20x20, 30x30, 50x50, 80x80 and 125x125 nodes meshes.

	Ra			
	<b>10<sup>3</sup></b>	<b>10<sup>4</sup></b>	<b>10<sup>5</sup></b>	<b>10<sup>6</sup></b>
$u_{Max}$	3.646	16.195	34.996	65.463
$z$	0.812	0.820	0.852	0.852
$v_{Max}$	3.700	19.650	68.678	222.068
$x$	0.180	0.116	0.068	0.036
$\bar{Nu}$	1.118	2.247	4.535	8.907
$Nu_{Simp}$	1.116	2.256	4.433	8.588
$Nu_{Max}$	1.507	3.537	7.771	18.075
$z$	0.092	0.140	0.084	0.036
$Nu_{Min}$	0.691	0.585	0.728	0.982
$z$	1	1	1	1

Table 20: Differentially Heated Cavity. More accurate obtained results (125x125 meh)



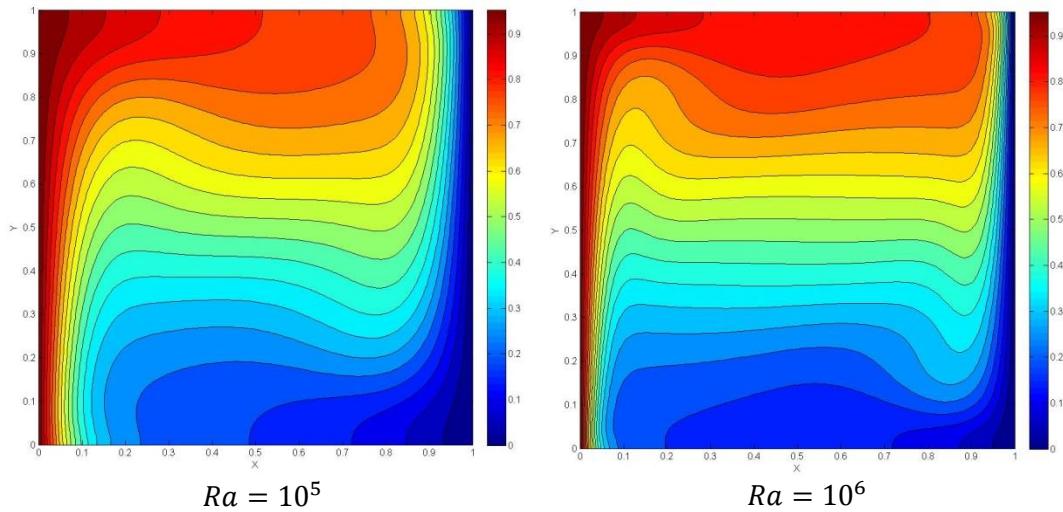


Figure 40: Differentially Heated Cavity. Isolines of the temperature field with a 125x125 mesh

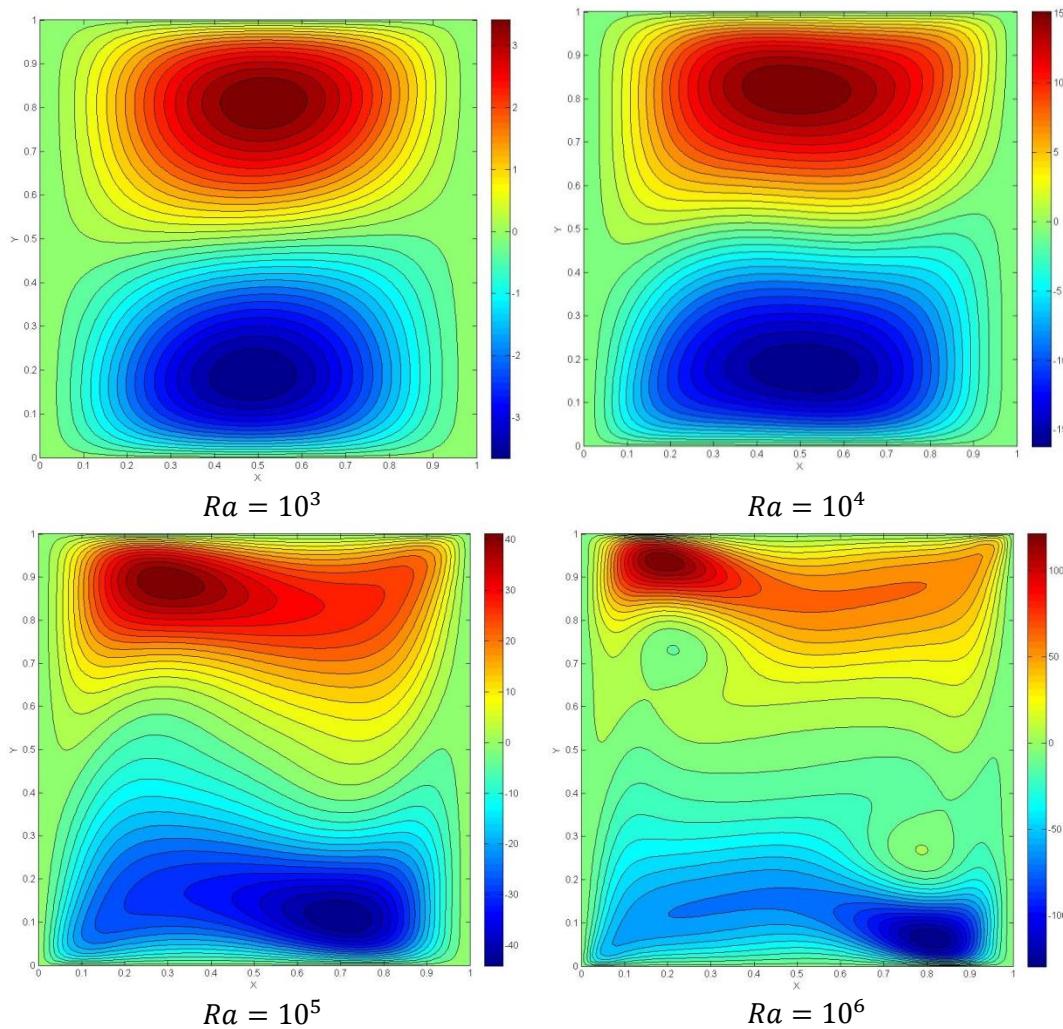


Figure 41: Differentially Heated Cavity. Isolines of the U component field with a 125x125 mesh

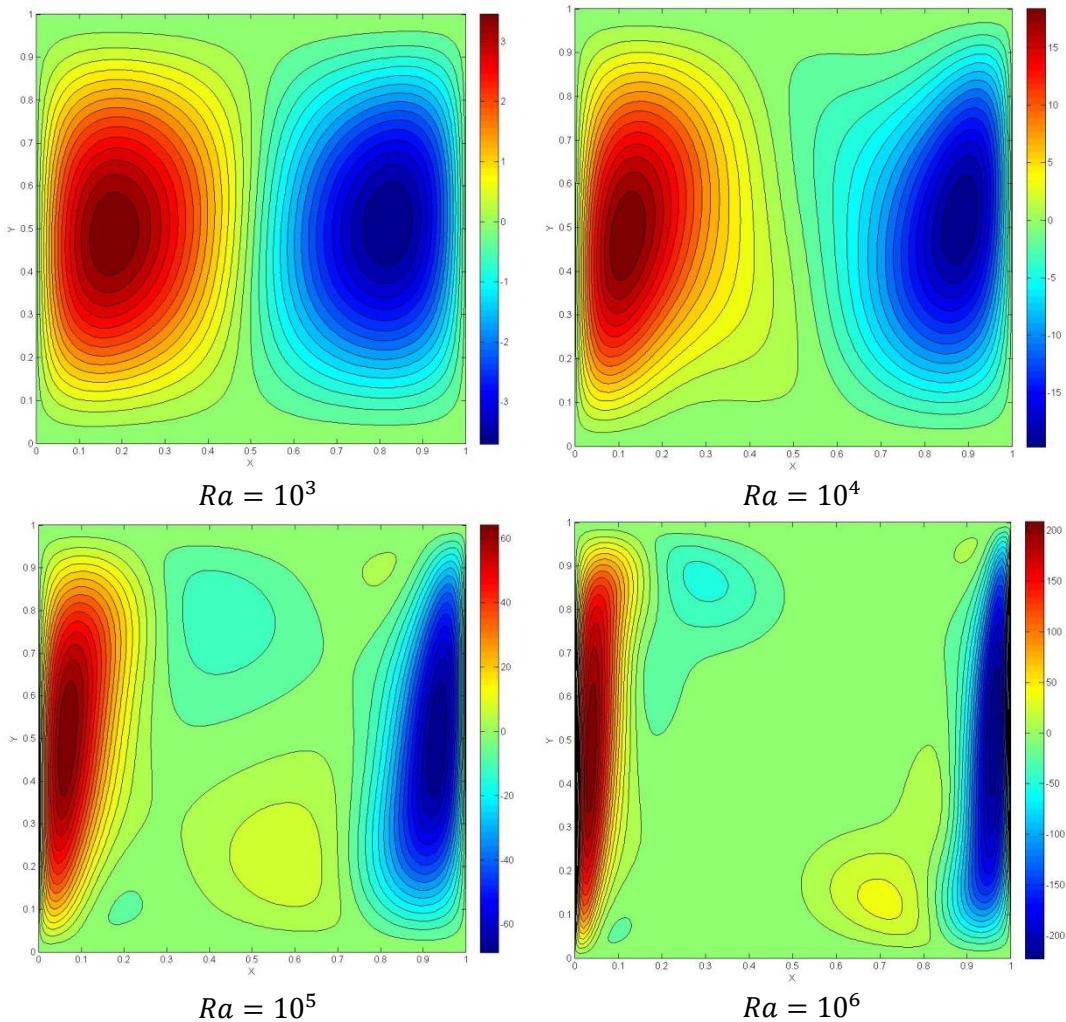
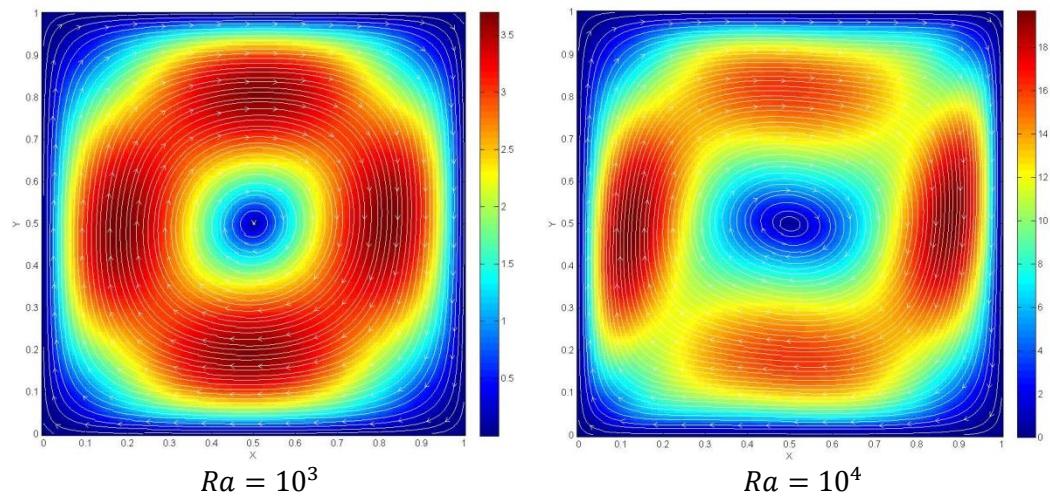


Figure 42: Differentially Heated Cavity. Isolines of the  $V$  component field with a 125x125 mesh



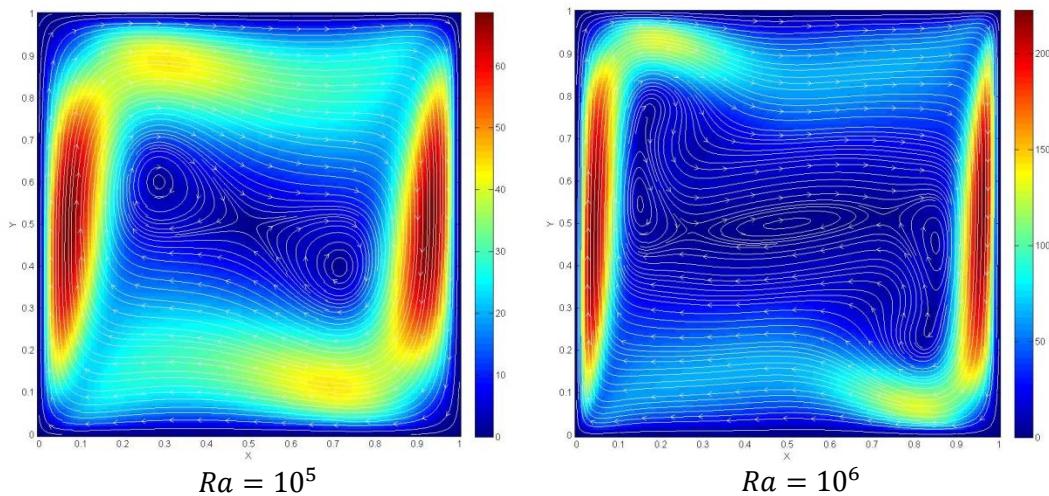


Figure 43: Differentially Heated Cavity. Velocity field and streamlines with a 125x125 mesh

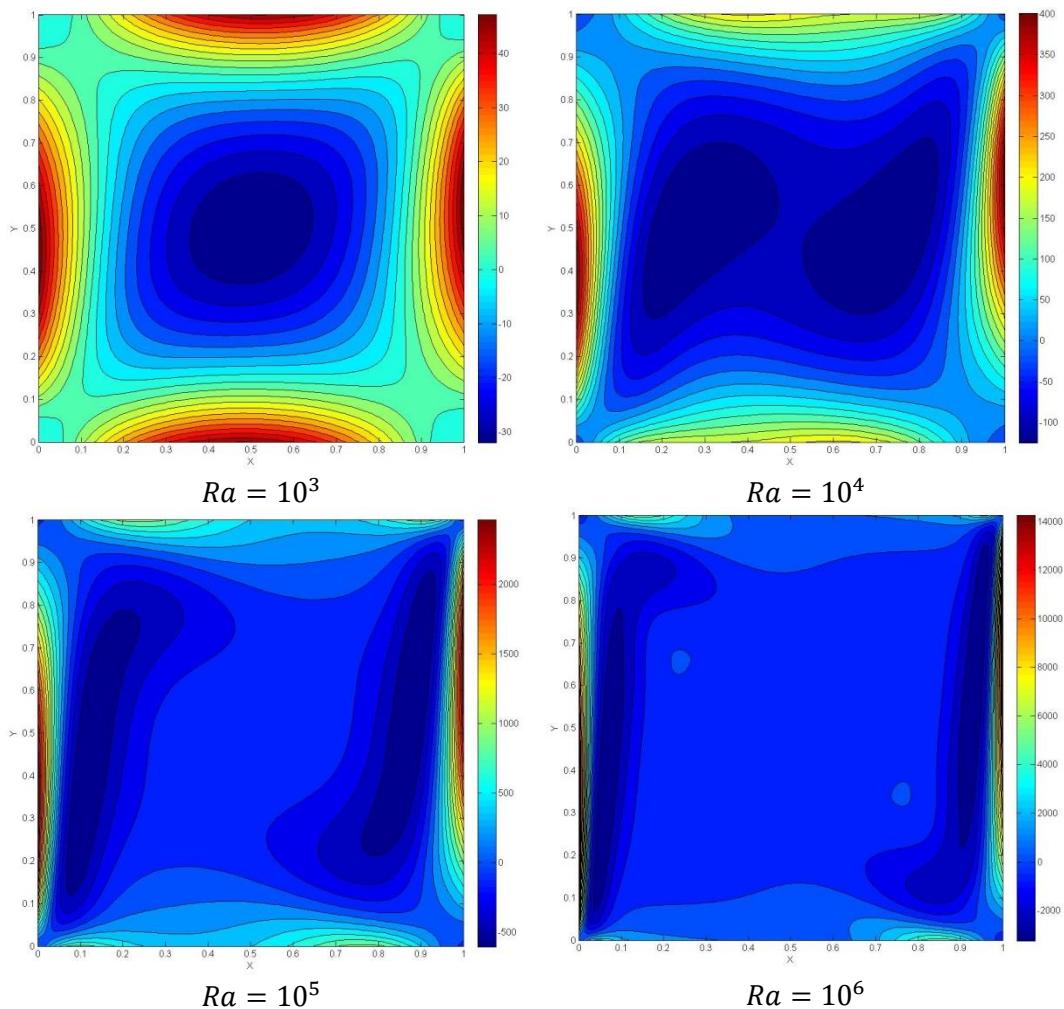


Figure 44: Differentially Heated Cavity. Isolines of the vorticity field with a 125x125 mesh

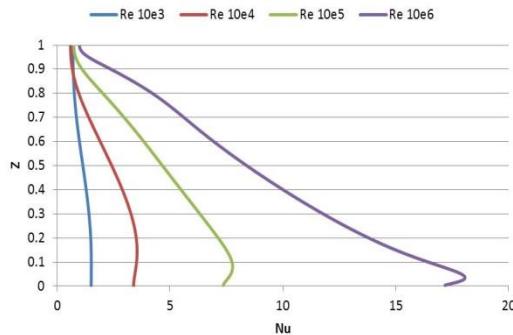


Figure 45: Differentially Heated Cavity. Nusselt number distribution at the hot wall for different Rayleigh numbers with a 125x125 mesh

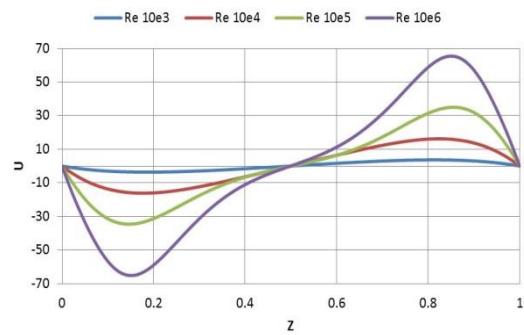


Figure 46: Differentially Heated Cavity. U velocity component in the vertical mid-plane for different Rayleigh numbers with a 125x125 mesh

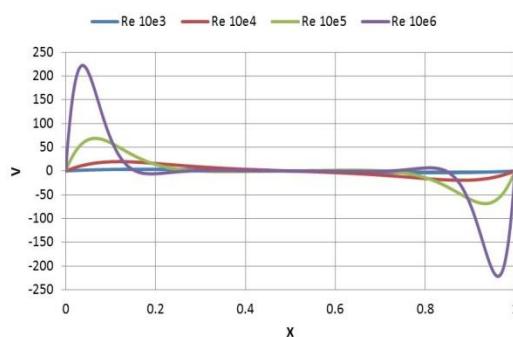


Figure 47: Differentially Heated Cavity. V velocity component in the horizontal mid-plane for different Rayleigh numbers with a 125x125 mesh (1)

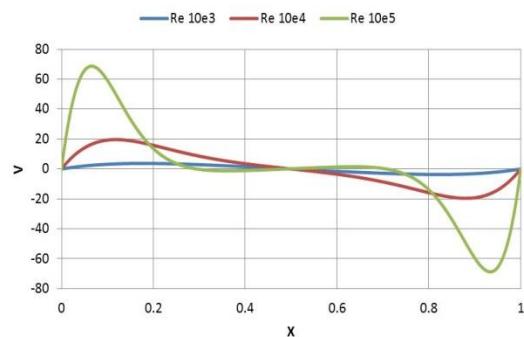


Figure 48: Differentially Heated Cavity. V velocity component in the horizontal mid-plane for different Rayleigh numbers with a 125x125 mesh (2)

### 2.6.2.5. Conclusions

This problem has computed the velocity field and the temperature field inside a square cavity filled with a Boussinesq fluid for a constant Prandtl number and four different Rayleigh numbers, which give the importance of the influence of the temperature gradient in the velocity field.

First of all, as seen in *Table 20*, the most accurate results are very similar to the reference ones for the low Rayleigh numbers, but as this number increases, the results start to differ but with a good trend to converge (see *Attachment 2*). This follows the previous lines seen in other validation problems that the code is not prepared for high velocity-gradient fields or turbulent flows. Then, more mesh refinement should be done in order to obtain better results with the developed code. When taking a look at the contour plots for the different cases, a good similarity with the reference plots has been observed (see [7]).

As a last point to comment, a comparison between different Rayleigh numbers can be seen in *Figures 45 to 48*. When the Rayleigh number increases, the influence of the temperature field to the velocity field becomes stronger. It can be seen how the V-component velocity becomes bigger near the left and right cavity walls and, also, how this behavior influences the U-component velocity near the top and the bottom wall, making the fluid rotate faster inside the cavity. Also a higher heat transfer on the left wall is observed for higher Rayleigh numbers with the Nu distribution.

### 2.6.3. Flow around a square cylinder

#### 2.6.3.1. Objective

This is a proposed problem by CTTC as an application of the Fractional Step Method. The objective of the problem is to solve the velocity field around a square cylinder and, then, compute aerodynamic coefficients like the drag, the Strouhal number for non-steady cases, etc. All the information regarding this case can be found in [10] and [11].

#### 2.6.3.2. Problem definition

This case is divided in two different problems to be solved. The first one consists in solving the flow around a square cylinder inside a plane channel for different Reynolds numbers and is presented in [10]. The other one is presented in [11] and consists in solving the interferences in the flow generated by two square cylinders (or a square cylinder array) with different distance between them for a fixed Reynolds number of 100. Due to the available time and computational cost, the first case could only be simulated with an equidistant mesh, used also in the reference solution (but with a different method), and the second one with a non-equidistant mesh only for the case of two square cylinders, meant more to obtain qualitative than quantitative results.

#### Case 1

The following figure shows the geometry of the plane channel:

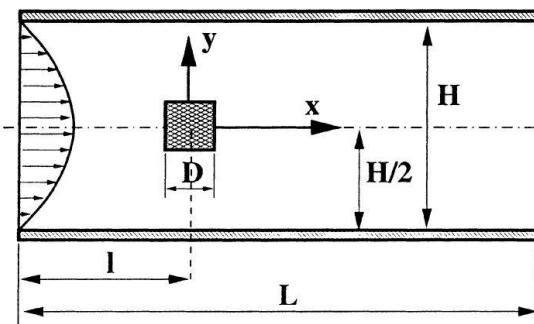


Figure 49: Square cylinder. Scheme of the problem (case 1) (extracted from [10])

The geometry is defined with the blockage ratio of the channel  $B = D/H = 1/8$ . Moreover, the length of the channel is set to  $L/D = 50$  in order to reduce the influence of the inflow and outflow conditions and the inflow length is set to  $l = L/4$ .

The boundary conditions are shown in the following equations:

$$\vec{v} = \vec{0}; \frac{\partial p}{\partial y} = 0; \text{ (top cavity wall)} \quad (137)$$

$$\vec{v} = \vec{0}; \frac{\partial p}{\partial y} = 0; \text{ (bottom cavity wall)} \quad (138)$$

$$u = \frac{1}{2}y - \frac{1}{16}y^2; v = 0; \frac{\partial p}{\partial x} = 0; \text{ (left cavity wall)} \quad (139)$$

$$\frac{\partial \vec{v}}{\partial \vec{x}} = \vec{0}; p = 0; \text{ (right cavity wall)} \quad (140)$$

Moving the origin of the axes of coordinates to the bottom-left corner, it is possible to define the parabolic inflow velocity profile as seen in equation (139) with the unity as its maximum value. Furthermore, in this case the pressure is imposed to be zero at the outflow in order to solve the Poisson equation.

The studied cases are for Reynolds numbers 1, 5, 10, 30, 50, 60, 100, 150 and 200 (which is defined using as a reference velocity the maximum velocity of the parabolic inflow profile ( $u_{Max}$ ) and as a reference distance the diameter D of the square cylinder).

The problem is studied using  $D = 1$  and  $u_{Max} = 1$  so the dimensions and variables can be treated as dimensionless variables.

### Case 2

The following figure shows the geometry of this case:

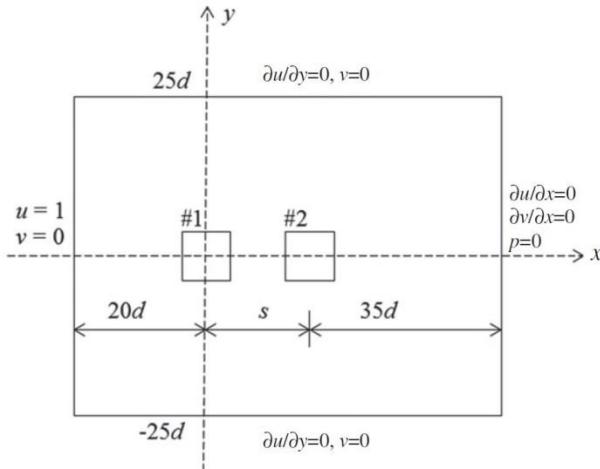


Figure 50: Square cylinder. Scheme of the problem (case 2) (extracted from [11])

In the previous figure,  $d$  refers to the diameter of the square cylinder and  $s$  refers, in this case, to the distance between the centers of the two square cylinders and it is defined with the separation ratio parameter  $s/d = <\text{Value}>$ . In this case the geometry also guarantees that the outflow and inflow boundary conditions will not be affected by each other.

The boundary conditions can be summarized as follows:

$$\frac{\partial u}{\partial y} = 0; v = 0; \frac{\partial p}{\partial y} = 0; \text{ (top cavity wall)} \quad (141)$$

$$\frac{\partial u}{\partial y} = 0; v = 0; \frac{\partial p}{\partial y} = 0; \text{ (bottom cavity wall)} \quad (142)$$

$$u = 1; v = 0; \frac{\partial p}{\partial x} = 0; \text{ (left cavity wall)} \quad (143)$$

$$\frac{\partial \vec{v}}{\partial \vec{x}} = \vec{0}; p = 0; \text{ (right cavity wall)} \quad (144)$$

In this case, the velocity takes a constant value in the entire left wall and the pressure is also imposed to be zero at the right wall in order to solve the Poisson equation.

The Reynolds number is 100 and is defined using the inflow velocity as a reference velocity and the diameter of the square cylinder as a reference distance. The problem is studied as before using  $D = 1$  and  $u_{Max} = 1$  so the dimensions and variables can be treated as dimensionless variables.

The studied cases correspond to s/d values of 1.5, 2, 3, 4, 5, 6, 7 and 8. Because of time limits and computational costs, the cases of s/d corresponding to 9, 10 and 11 have not been simulated.

#### 2.6.3.3. Numerical discretization

This problem returns to the origins of the Driven Cavity problem, but with the difference that the obstacle must be included inside the system of equations and that the boundary conditions of the pressure field are a little bit different.

The boundary nodes of the pressure field are discretized as follows:

Top cavity wall	Value	Bottom cavity wall	Value	Upper-left corner	Value	Upper-right corner	Value
$a_P$	1	$a_P$	1	$a_P$	1	$a_P$	1
$a_E$	0	$a_E$	0	$a_E$	0.5	$a_E$	0
$a_W$	0	$a_W$	0	$a_W$	0	$a_W$	0.5
$a_N$	0	$a_N$	1	$a_N$	0	$a_N$	0
$a_S$	1	$a_S$	0	$a_S$	0.5	$a_S$	0.5
$b_P$	0	$b_P$	0	$b_P$	0	$b_P$	0
Left cavity wall	Value	Right cavity wall	Value	Lower-left corner	Value	Lower-right corner	Value
$a_P$	1	$a_P$	1	$a_P$	1	$a_P$	1
$a_E$	1	$a_E$	0	$a_E$	0.5	$a_E$	0
$a_W$	0	$a_W$	0	$a_W$	0	$a_W$	0.5
$a_N$	0	$a_N$	0	$a_N$	0.5	$a_N$	0.5
$a_S$	0	$a_S$	0	$a_S$	0	$a_S$	0
$b_P$	0	$b_P$	0	$b_P$	0	$b_P$	0

Table 21: Square cylinder. Discretization coefficients of boundary nodes of the pressure field

Then, in order to include the presence of the obstacles inside our domain, the velocity nodes corresponding to the square cylinder are set to zero (no-slip boundary condition) and for the inner nodes of the pressure field, the control volumes inside the obstacle are set to zero and those who

touch the walls of the square cylinder have its directional coefficient pointing to the obstacle set to zero. It means that no pressure flow is transferred to the obstacle. This is done setting, for example, the  $a_E$  east coefficient to zero when the control volume is touching the obstacle with its east face.

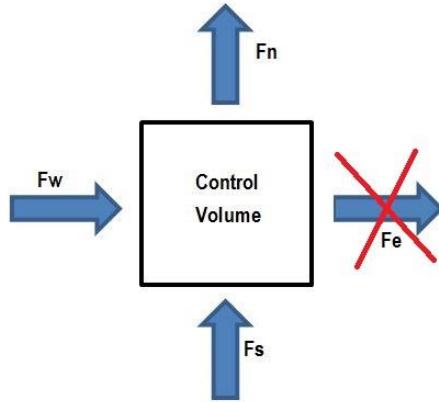


Figure 51: Square cylinder. Representation of a control volume touching the obstacle with its east face

The most relevant parameters to be computed in these cases are the aerodynamic coefficients of the square cylinder and the Strouhal number in non-steady cases.

These parameters are defined as follows (see [11]):

$$C_l = \frac{F_l}{\frac{1}{2} \rho u_{Max}^2 D} \quad (145)$$

$$C_d = \frac{F_d}{\frac{1}{2} \rho u_{Max}^2 D} \quad (146)$$

$$St = \frac{fD}{u_{Max}} \quad (147)$$

The aerodynamic coefficients are computed according to [12].

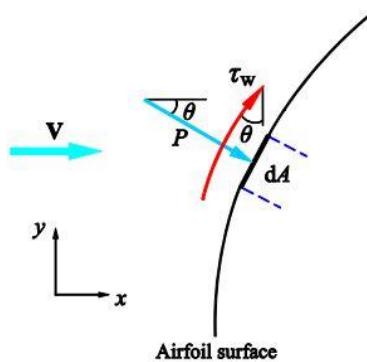


Figure 52: Square cylinder. Aerodynamic forces on an elemental area (extracted from [12])

$$F_l = \int_A dF_y = - \int_A p \sin \theta dA + \int_A \tau_w \cos \theta dA \quad (148)$$

$$F_d = \int_A dF_x = \int_A (pdA) \cos \theta + \int_A (\tau_w dA) \sin \theta \quad (149)$$

This integration can be done making a sum of all the discrete forces obtained on each control volume touching the obstacle. In order to obtain the pressure, it must be recovered from the pseudo-pressure as  $p^{n+1} = \tilde{p}/\Delta t$ . Then, the pressure is interpolated to the wall of the obstacle, since it is computed in the middle of the control volume. Furthermore, to compute the stress coefficient it is necessary to follow the evaluation of the diffusive term in the Navier-Stokes equations (equation (152)) (see [13]).

$$\nabla \cdot \vec{\tau} = \mu \nabla^2 \vec{u} \quad (150)$$

$$\vec{\tau} = \mu (\nabla \vec{u} + (\nabla \vec{u})^T) \quad (151)$$

$$\tau = \mu \begin{pmatrix} 2 \frac{\partial u}{\partial x} & \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \\ \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} & 2 \frac{\partial v}{\partial y} \end{pmatrix} \quad (152)$$

Then, the tangential stress coefficient is needed, which corresponds to  $\tau_w = \mu \left( \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right)$ . This value can be computed numerically on a pressure node as:

$$\tau_w = \mu \left( \frac{u_n - u_s}{\Delta y} + \frac{v_e - v_w}{\Delta x} \right) \quad (153)$$

To compute all these previous parameters a reference density must be defined and, using equation (108), the dynamic viscosity can be found.

The Strouhal number is computed from a Power Spectral Density (PDS) analysis (using a Fast Fourier Transformation - FFT) of time series of the lift coefficient (see [14] for more information). Since the reference distance and the reference velocity are equal to one, the Strouhal number is directly the frequency obtained from this PDS.

Furthermore, for the second case a non-equidistant mesh has been implemented in order to reduce the computation time since the simulation domain is much bigger than the obstacles zone. This mesh consists in an equidistant separation in the x and y direction between the obstacles and the obstacles themselves and a non-equidistant mesh from the obstacles until the boundaries of the domain, which follows the law:

$$h_i = \begin{cases} i ; 1 \leq i \leq n/2 \\ n - i ; n/2 \leq i \leq n \end{cases} \quad (154)$$

$$d_{i+1} = \begin{cases} 0 ; i = 1 \\ \frac{\sum_1^{i-1} h_i}{\sum_1^n h_i} ; 2 \leq i \leq n \end{cases} \quad (155)$$

Where  $n$  is the double of the number of nodes that are set in the non-equidistant zones,  $d$  its position in the domain and  $h$  corresponds to the control volume size. These equations generate a mesh that starts being thin, then coarse and then thin again, so only the half of this generated mesh is taken, putting the thinnest nodes near the obstacles.

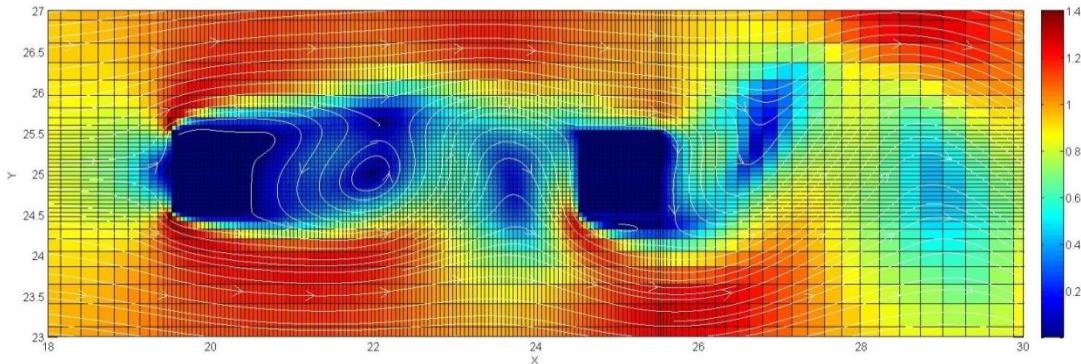


Figure 53: Square cylinder. Non-equidistant mesh used in the case 2 (close view near the obstacles)

To close this section, the global algorithm of resolution is the same as the one explained in the Driven Cavity problem (section 2.6.1.4).

#### 2.6.3.4. Results

##### Case 1

The following results are the most relevant obtained for this case. For the steady cases, the following figures show the steady drag coefficient and the recirculation length of the vortexes generated behind the obstacle (made dimensionless with the diameter of the square cylinder). For the non-steady cases (from  $Re = 60$ ), the time-averaged drag coefficient, the difference between the maximum and minimum drag and lift coefficients and the Strouhal number are compared with the reference values. All the reference values have been digitized from [10] from the case of FVM EQ 500x80, which is also the mesh used in the simulations of this problem.

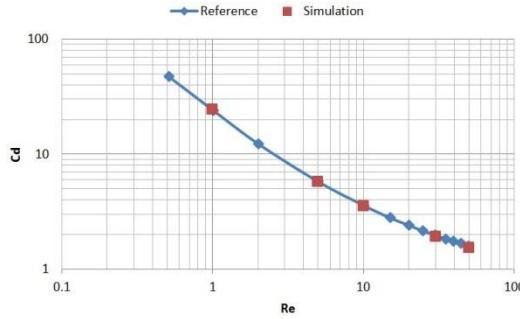


Figure 54: Square cylinder (case 1). Evolution of the steady drag coefficient with the Reynolds number

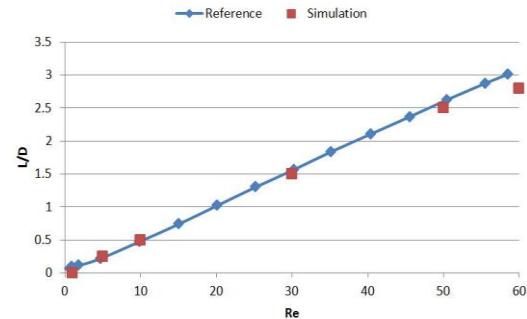


Figure 55: Square cylinder (case 1). Evolution of the recirculation length with the Reynolds number

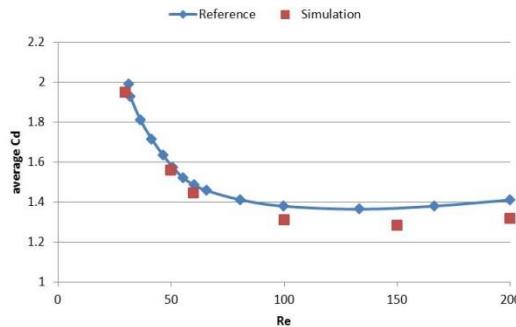


Figure 56: Square cylinder (case 1). Evolution of the time-averaged drag coefficient with the Reynolds number

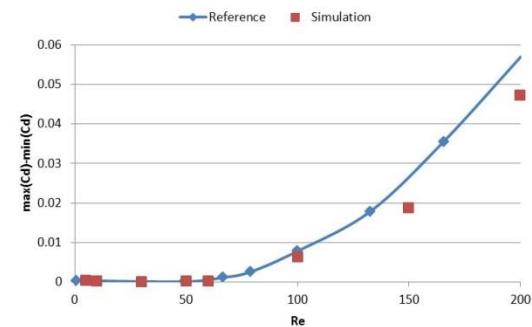


Figure 57: Square cylinder (case 1). Evolution of the difference between maximum and minimum drag coefficient with the Reynolds number

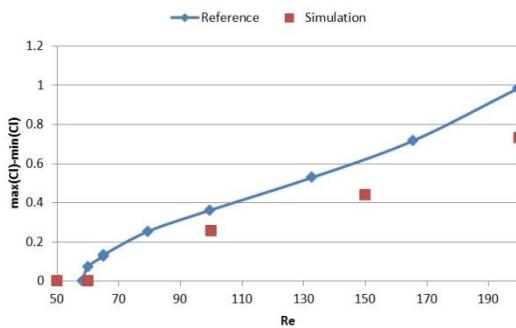


Figure 58: Square cylinder (case 1). Evolution of the difference between maximum and minimum lift coefficient with the Reynolds number

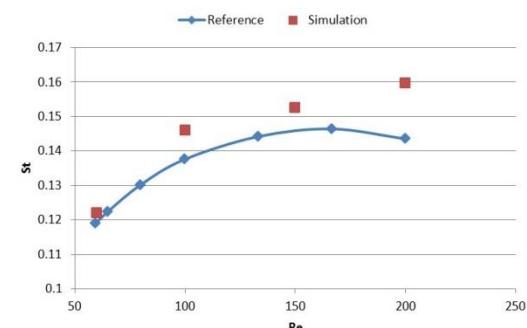


Figure 59: Square cylinder (case 1). Evolution of the Strouhal number with the Reynolds number

Now, the streamlines around the square cylinder are shown as qualitative results (but from which the recirculation length was extracted) to compare with those reference figures shown in [10]. The figures showing the Reynolds numbers from 1 to 50 are steady cases, while the figures showing the Reynolds numbers 100 and 200 are instantaneous plots of the streamlines.

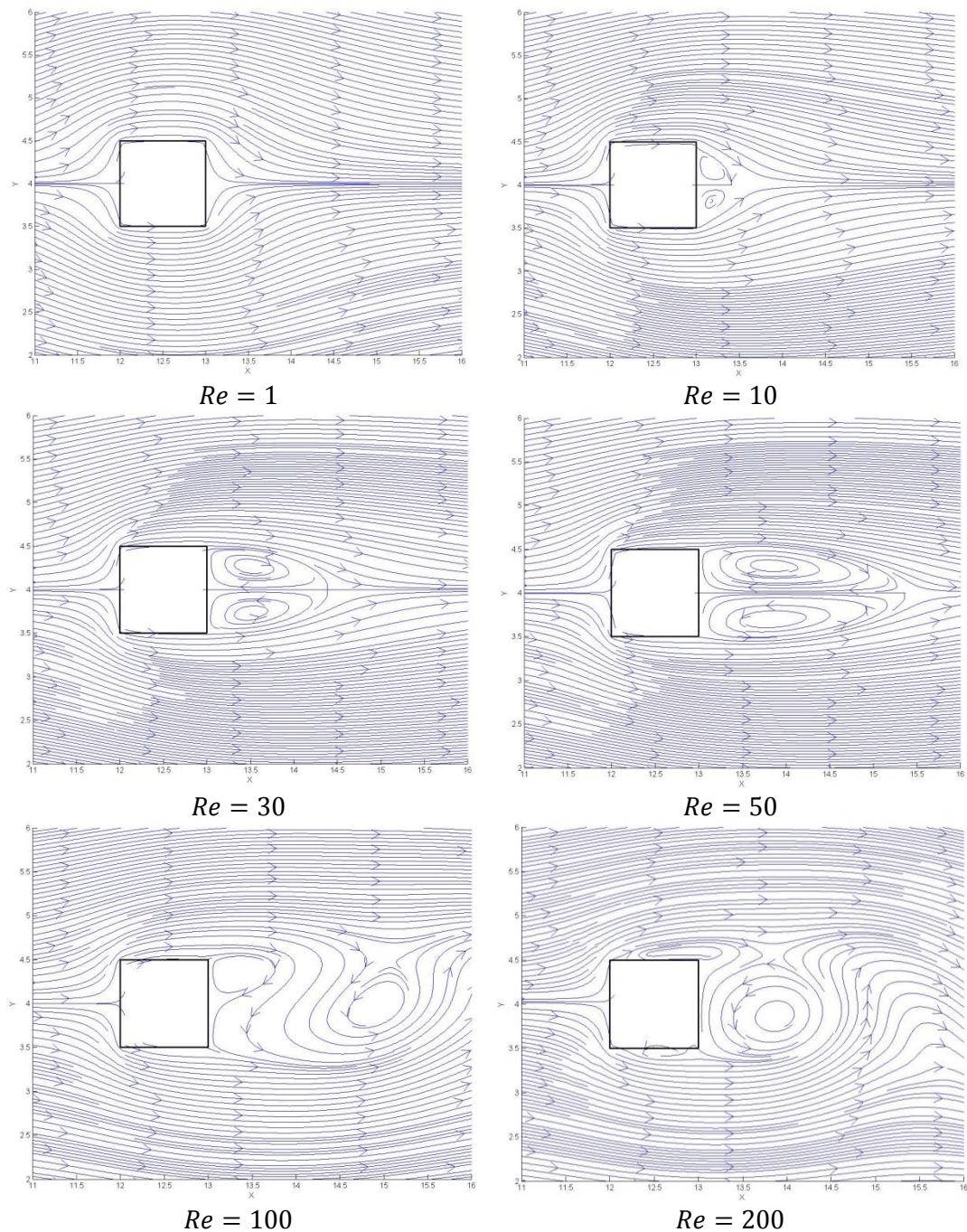


Figure 60: Square cylinder (case 1). Streamlines around the obstacle for different Reynolds numbers

The following figures show the velocity field and the pressure field for different Reynolds numbers, as a qualitative result.

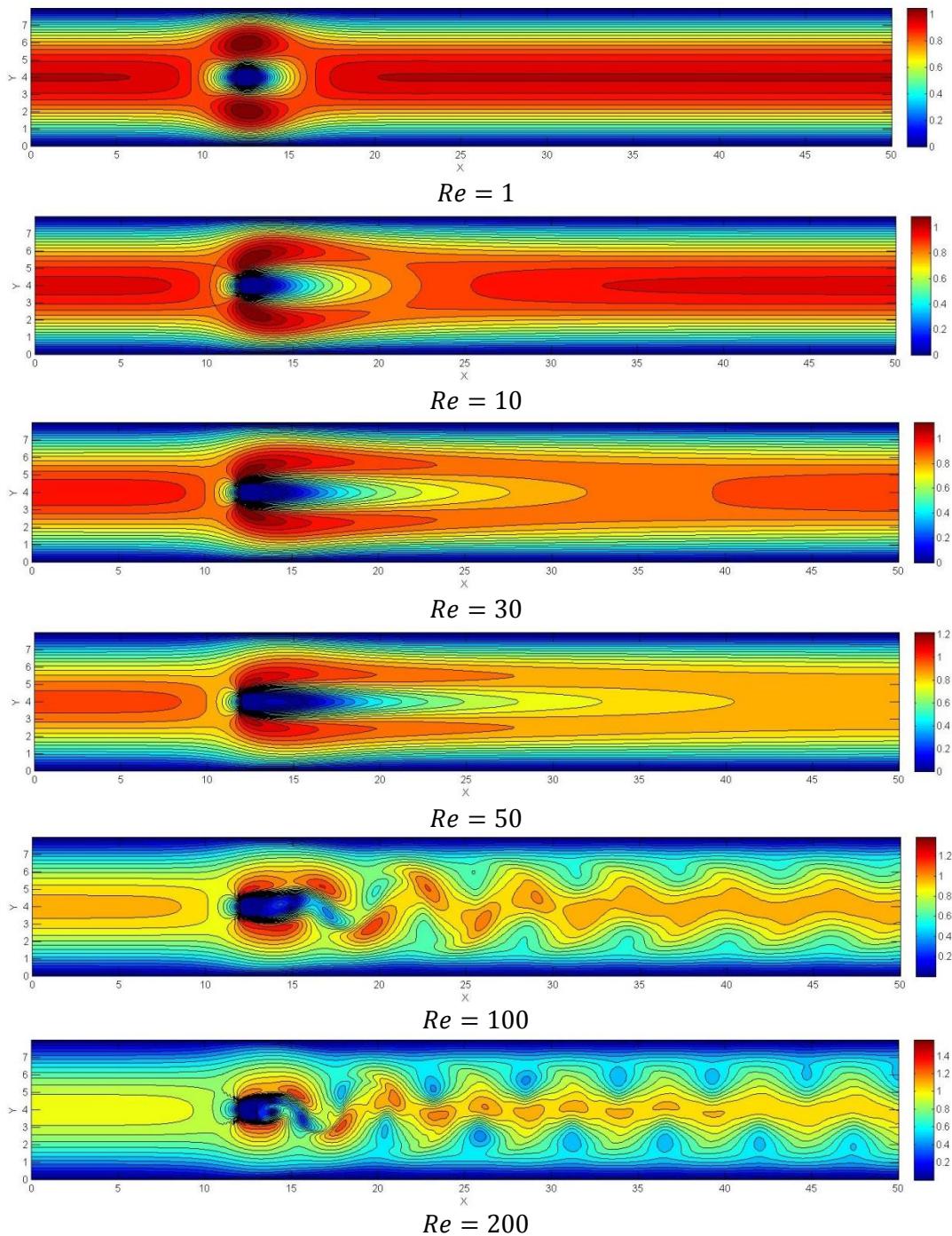
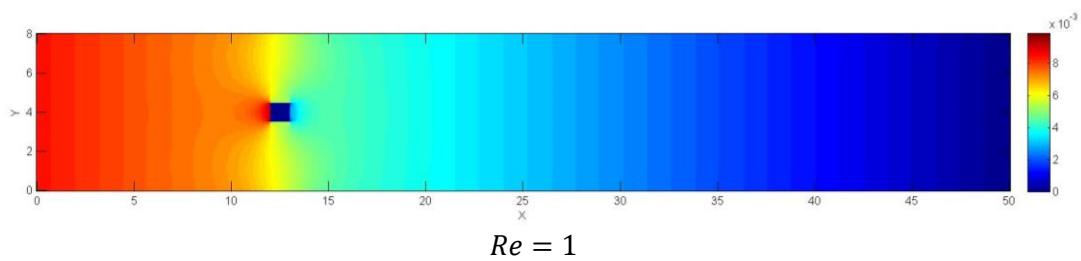


Figure 61: Square cylinder (case 1). Isolines of the velocity field for different Reynolds numbers



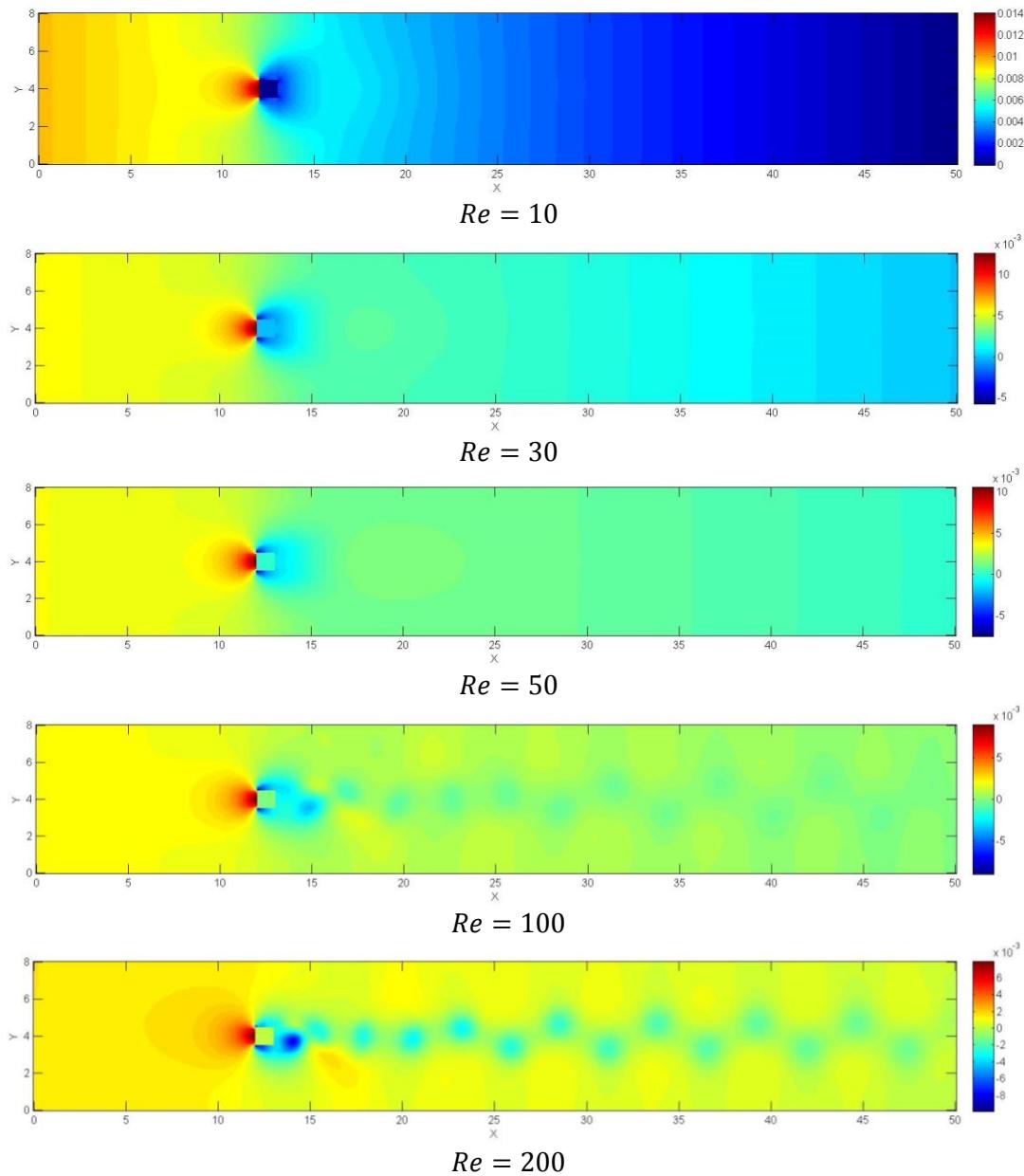


Figure 62: Square cylinder (case 1). Pseudo-pressure field for different Reynolds numbers

### Case 2

The following mesh sizes have been used to solve the different situations. All of them have 20 control volumes for each side of the square cylinder, instead of 10 as the equidistant mesh of the previous case.

s/d	Mesh size	s/d	Mesh size
1.5	170x100	5	240x100
2	180x100	6	260x100
3	200x100	7	280x100
4	220x100	8	300x100

Table 22: Square cylinder (case 2). Mesh sizes according to s/d ratio

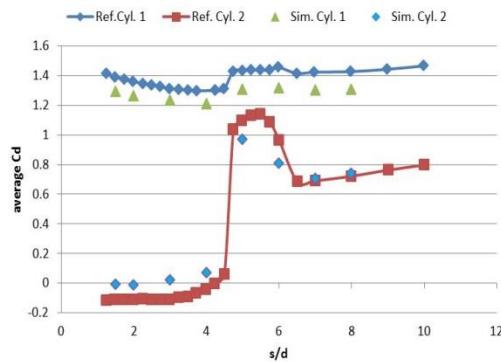


Figure 63: Square cylinder (case 2). Evolution of the time-averaged drag coefficient with the separation ratio  $s/d$

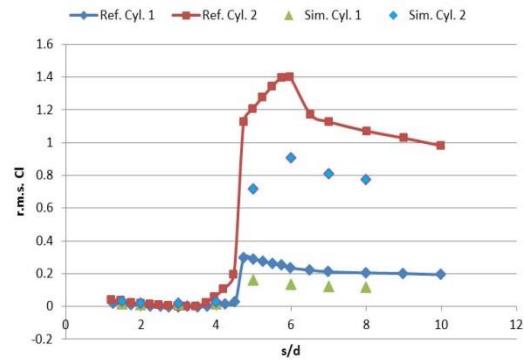


Figure 64: Square cylinder (case 2). Evolution of the root mean square value of the lift coefficient with the separation ratio  $s/d$

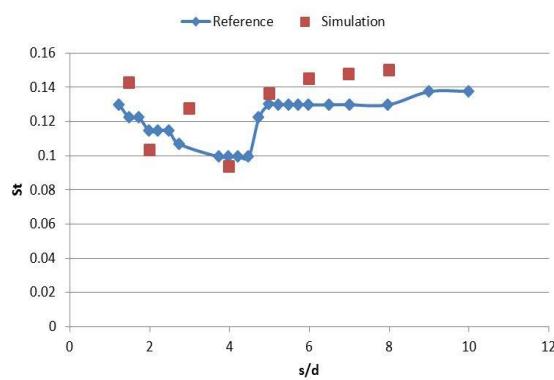
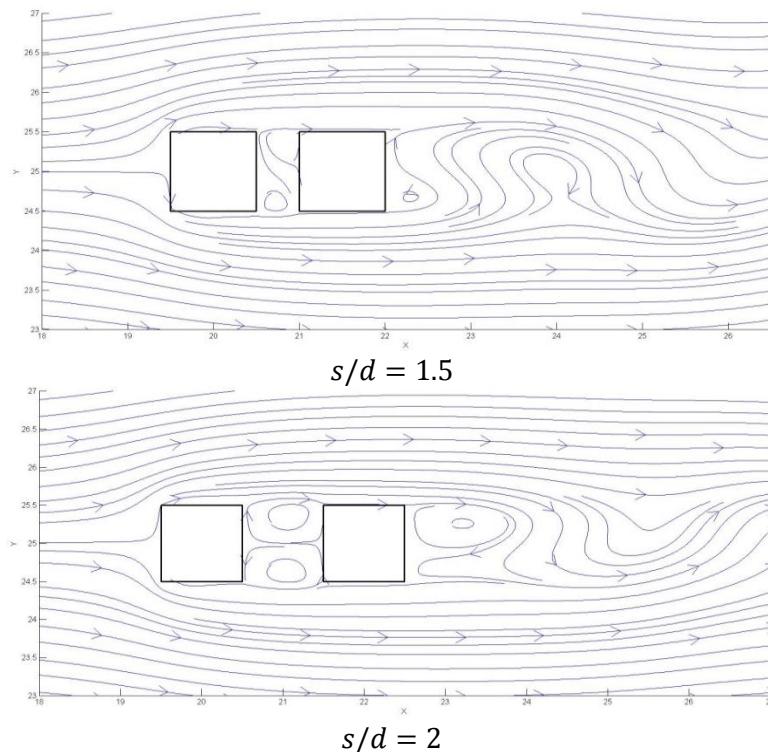
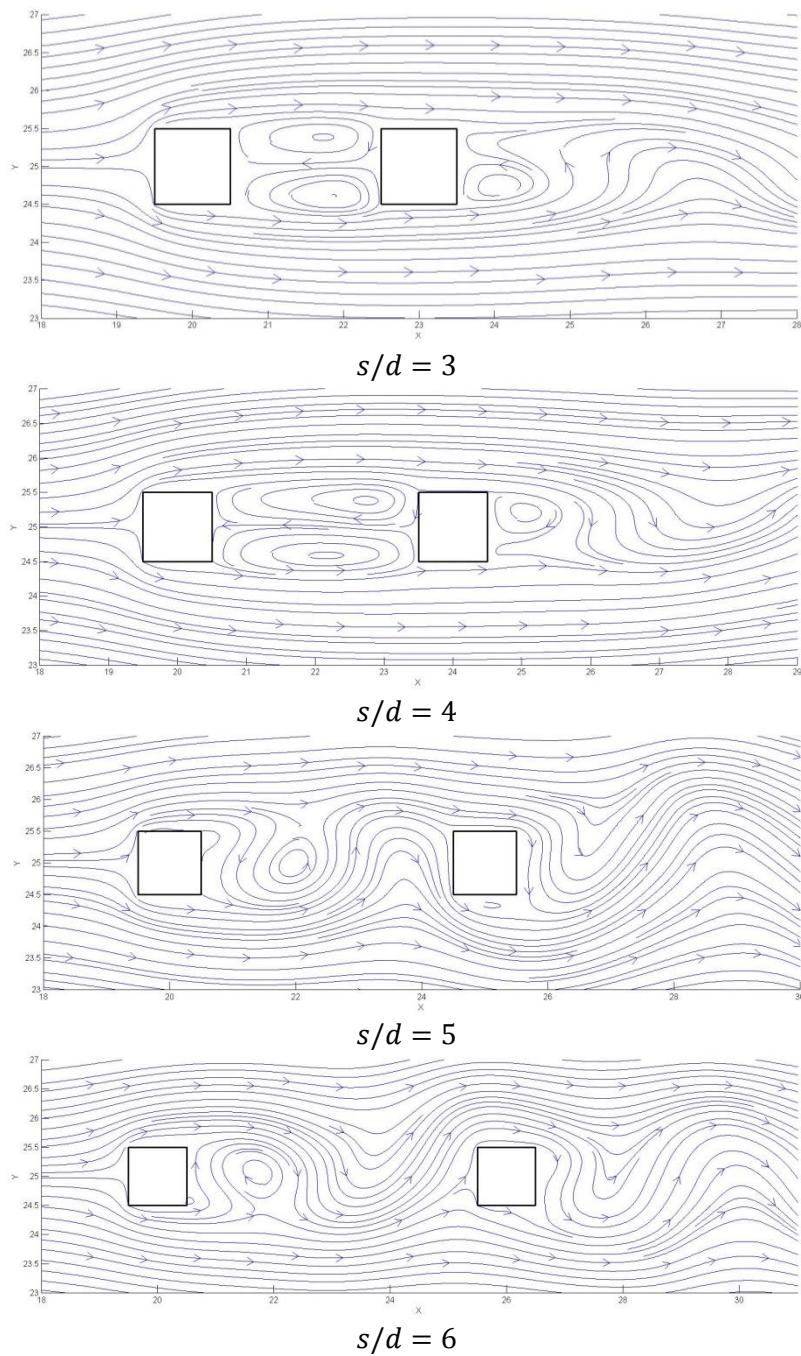
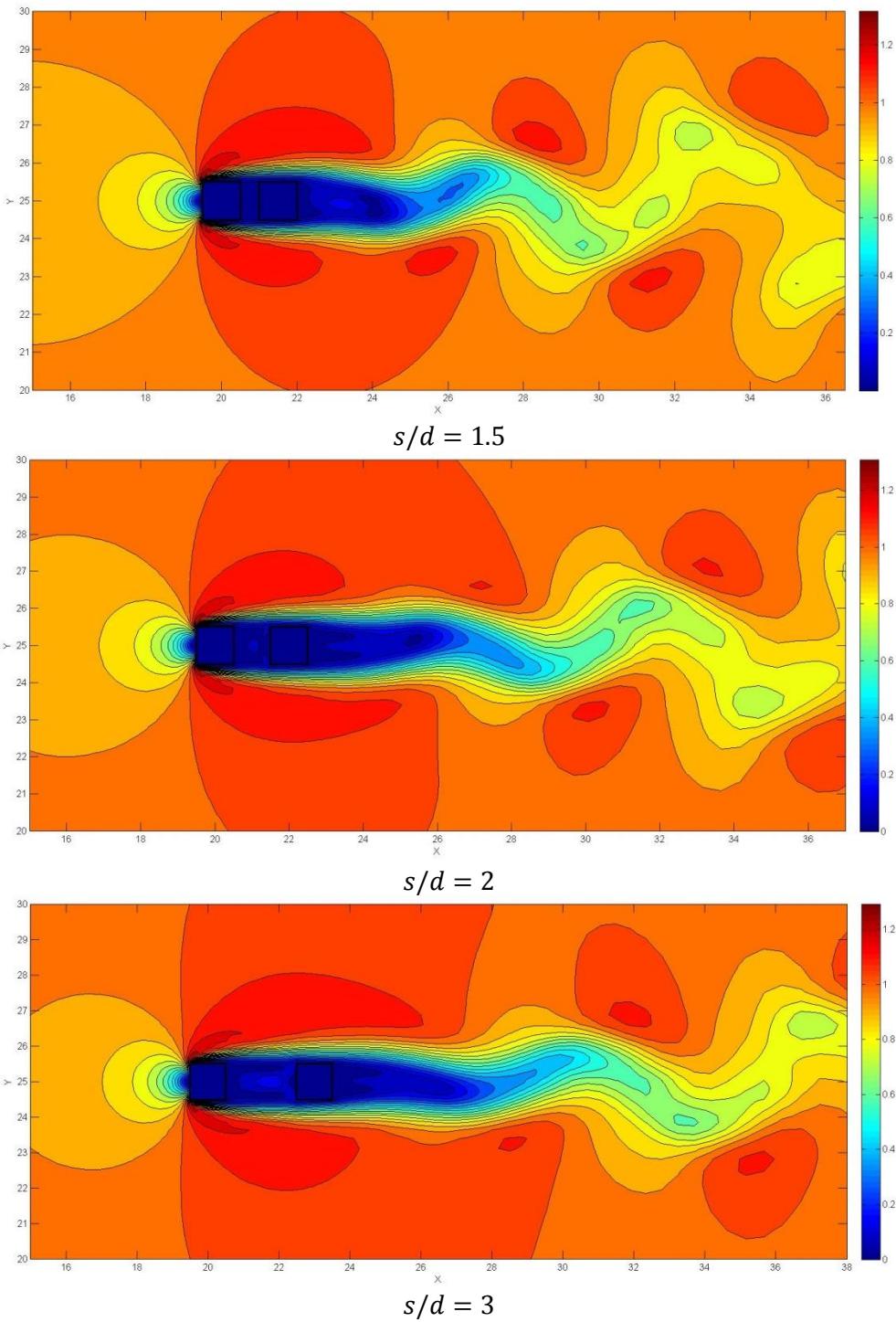


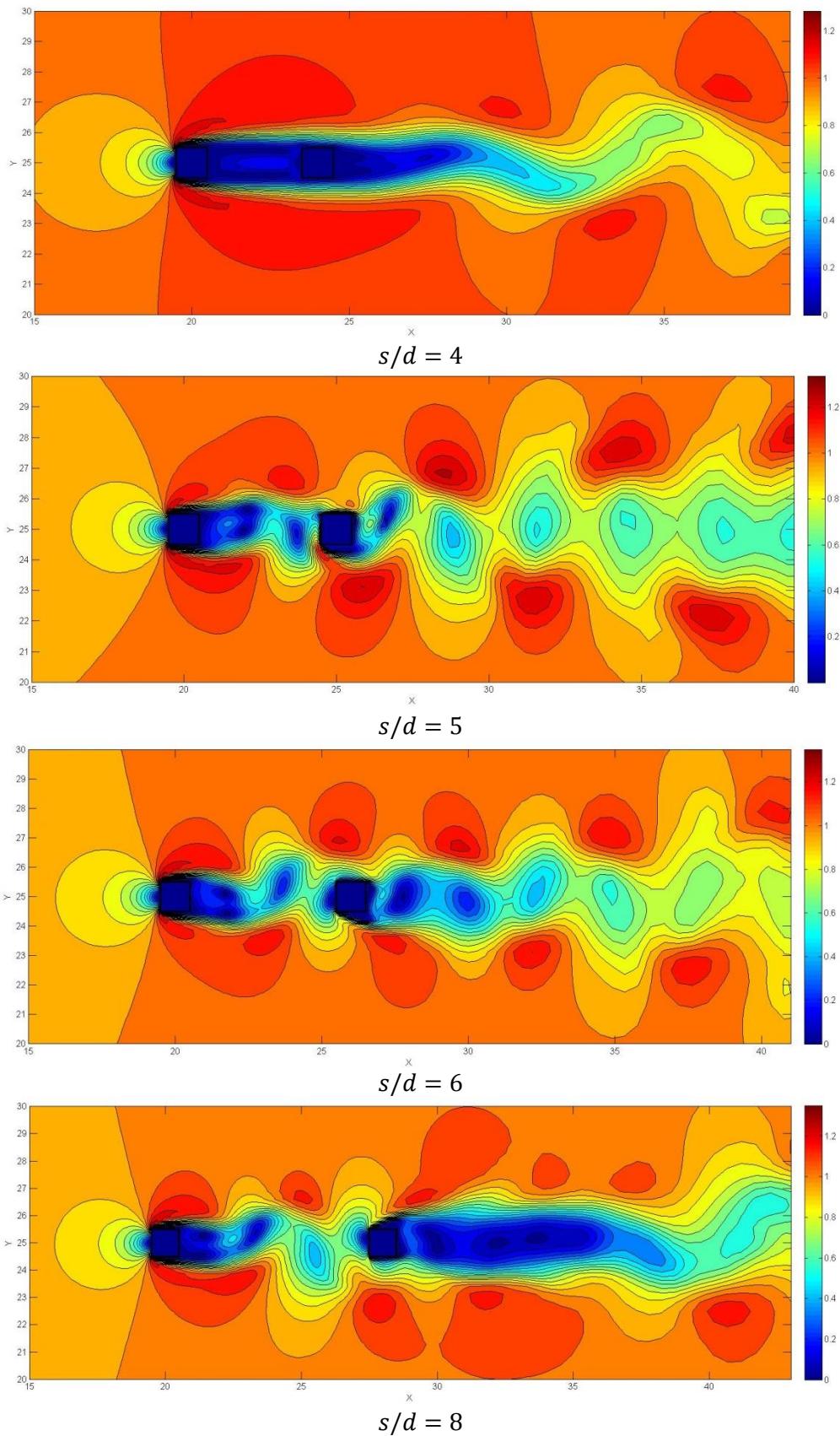
Figure 65: Square cylinder (case 2). Evolution of the Strouhal number with the separation ratio  $s/d$





**Figure 66: Square cylinder (case 2). Evolution of instantaneous streamlines around the two obstacles with the separation ratio  $s/d$**





**Figure 67: Square cylinder (case 2). Evolution of instantaneous velocity field around the two obstacles with the separation ratio  $s/d$**

### 2.6.3.5. Conclusions

This problem is the first one applied to an aerodynamic field, giving the opportunity to compute aerodynamic parameters such as drag coefficients or the Strouhal number of the non-steady oscillations.

The first case shows the evolution of the velocity field around a square cylinder while increasing the Reynolds number of the fluid. Good concordances with the reference results are observed (*Figures 54 to 59*) for the low Reynolds and steady cases, but as this number increases and the flow becomes more turbulent and non-steady, the obtained results start to deviate from the reference ones due to the developed code, which is not ready to solve properly the turbulence without a really fine mesh. This transition occurs according to [10] at a critical Reynolds number of 54 (at least it should be lower than 70). In our case, the simulation is still steady for  $Re = 60$ , but taking very much time to converge. With a finer mesh, the critical Reynolds would be closer to 54.

In the *Figure 60* it is possible to observe the evolution of the streamlines around the obstacle while increasing the Reynolds number (it also corresponds to the reference solutions given in [10]). For low Reynolds numbers, the fluid can be attached to all the surface of the obstacle ( $Re = 1$ ) and, afterwards, only exists detachment of the fluid behind the obstacle, generating some recirculating vortexes which increase its length with the Reynolds number ( $Re = 10$ ,  $Re = 30$  and  $Re = 50$ ) until the critical value is reached and the flow starts fluctuating behind the square cylinder ( $Re = 100$  and  $Re = 200$ ). In these last two cases, there is also detachment of the fluid in the upper and lower walls of the obstacle, which becomes stronger as the Reynolds number increases. All this behavior can be also observed in *Figure 61*, which plots the isolines of the velocity field for different Reynolds numbers and a fluctuation of the fluid is seen for  $Re = 100$  and  $Re = 200$ . In *Figure 62*, the evolution of the pseudo-pressure field is shown. With these two figures, another conclusion can be extracted: once the flow is unsteady and oscillates behind the obstacle, these fluctuations take longer to dissipate as the Reynolds number increases.

The second case tried to investigate the interferences between two square cylinders inside a fluid with a constant Reynolds number 100 according to the separation ratio between them. In this case, even trying with a non-equidistant mesh which refined the control volumes near the obstacles, the numerical results differ considerably (*Figures 63 to 65*) with respect to the reference results presented in [11], but a common trend is observed between them. A refinement of the mesh would have given better results, but considering the computational costs and that for  $Re = 100$  there appears a large amount of motion scales (turbulence) and that the code is not prepared properly to solve these cases, it has not been done. Then, a more qualitative interpretation of the results has been done. It could also be possible that some error exists in the code when working with non-equidistant mesh and this fact should be considered as future work about this study.

For low separation ratios ( $s/d$ ), two recirculation vortexes appear between the two obstacles and cause the drag in the second obstacle to be small and the drag in the first obstacle to decay. For a critical separation ratio between 4 and 5, these two vortexes explode and the second obstacle finds

itself in the middle of the wake generated by the first one. This fact causes the drag and the root mean square value of the lift coefficient to increase drastically. Also the drag of the first obstacle increases a little bit since it is freed from the influence of the second square. These behaviors can be observed in the *Figures 66 and 67*.



### 3. INTRODUCTION TO TURBULENCE

The results obtained as a validation for the developed code showed how it was not prepared to solve properly turbulent cases, where big velocity gradients and small scales of motion appear produced by the convective term. The only available solution is to increase the mesh refinement, which is not recommended due to the increase of the computational costs.

The Navier-Stokes equations seen in the previous pages provide also an appropriate model for solving the nonlinear dynamics of turbulence, but a direct simulation (DNS) in turbulent cases is very difficult and expensive in terms of computational cost (basically, simulation time) as commented before. To have an idea of the smallest time/space scale to be solved, scaled with the Reynolds number, Kolmogorov (see [15]) proposed the following equivalences:

$$\delta t \sim Re^{-\frac{1}{2}} \quad (156)$$

$$\delta x \sim Re^{-\frac{3}{4}} \quad (157)$$

Then, the DNS memory requirements grow up to  $\sim Re^{\frac{9}{4}}$  (in 3D) and the computational cost to  $\sim Re^{\frac{11}{4}}$ .

In order to do a quick study of turbulence solvers, it is proposed to study the Burguers equation (158), which shares many of the aspects of the Navier-Stokes equations, but in an unidimensional domain (study proposed in [16]).

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} = \frac{1}{Re} \frac{\partial^2 u}{\partial x^2} + f \quad (158)$$

#### 3.1. Burguers equation in Fourier space

Considering the Burguers equation on an interval  $\Omega$  with periodic boundary conditions, the equation in Fourier space reads as:

$$\frac{\partial \hat{u}_k}{\partial t} + \sum_{k=p+q} \hat{u}_p i q \hat{u}_q = -\frac{k^2}{Re} \hat{u}_k + F_k \quad \text{where } k = 0, \dots, N \quad (159)$$

The forcing term is given by  $F_k = 0$  for  $k > 1$  and  $F_1$  such that  $d\hat{u}_1/dt = 0$  for  $t > 0$ . Then, the term  $\sum_{k=p+q} \hat{u}_p i q \hat{u}_q$  is the convective term and  $-\frac{k^2}{Re} \hat{u}_k$  the diffusive term. Also, the term  $\hat{u}_k \in \mathbb{C}$  corresponds to the  $k$ -th Fourier coefficient of  $u(x, t)$ .

$$u(x) = \sum_{k=-N}^{k=N} \hat{u}_k e^{ikx} \quad (160)$$

Then, it is important to remark that  $u(x, t) \in \mathbb{R}$ , so the condition  $\hat{u}_k = \overline{\hat{u}_{-k}}$  must be accomplished. This means that the  $k$ -th Fourier coefficient must be equal to its complex conjugate.

At this point, it is necessary to comment the role of each term. First of all, the convective term is transporting the energy from large scales (or low-frequency modes) to small scales (high-frequency modes) and also from small to large scales, since there is an energy backscattering. Therefore, on a DNS if the number of simulated scales is not sufficient, the results for the large scales will not have sense.

To continue, the diffusive term is damping energy or trying to eliminate the generated scales by the convective term. This damping effect is more effective for the high-frequency modes (small scales). The forcing term is the responsible of keeping the system energy (if not, it would dissipate). These forces are done in the large scales: smaller modes are easier to energize.

The last term to be analyzed is the Reynolds number. Increasing it, it yields to a decrease of the influence of the diffusive term. Then, the range of influence of the convective term is increased and more modes and scales are obtained. Because of this, with higher Reynolds numbers, a finer mesh is needed to fully solve the equation (to be related with the turbulent behavior).

In order to reduce the computational costs it is easier to implement a LES (Large-Eddy Simulation) model, which helps to obtain better results with a coarse mesh than with DNS. The Smagorinsky model [17] is the simplest LES model, which was released in the mid-60s. However, this model cannot be applied in Fourier space and a spectral eddy-viscosity model is proposed (see [16]).

In this model, the function  $\nu_t(k)$  from the Smagorinsky model is determined assuming some *a priori* properties of the energy spectrum. Taking into account the equations proposed by Métais and Lesieur [18], it is possible to reproduce energy spectra with different slopes.

$$\nu_t(k/k_N) = \nu_t^{+\infty} \left( \frac{E_{k_N}}{k_N} \right)^{\frac{1}{2}} \nu_t^* \left( \frac{k}{k_N} \right) \quad (161)$$

$$\nu_t^{+\infty} = 0.31 \frac{5-m}{m+1} \sqrt{3-m} C_K^{-\frac{3}{2}} \quad (162)$$

Here  $m$  represents the slope of the energy spectrum,  $E_{k_N}$  is the energy at the cutoff frequency ( $k_N$ ) and  $C_K$  is the Kolmogorov constant.  $\nu_t^*$  is a non-dimensional eddy-viscosity equal to 1 for small values of  $k/k_N$  and with a strong increase for higher  $k$  up to  $k/k_N = 1$ .

$$\nu_t^* \left( \frac{k}{k_N} \right) = 1 + 34.5 e^{-3.03 \left( \frac{k_N}{k} \right)} \quad (163)$$

For the case of the 1D Burgers equation, the slope of the energy spectrum is approximately  $m = 2$  and the Kolmogorov constant is  $C_K \approx 0.4523$  (see [16]). The energy refers to the kinetic energy, defined as:

$$E_k = \hat{u}_k \cdot \overline{\hat{u}_k} \quad (164)$$

At the end, this method consists in adding an extra viscosity, which increases with the modes, in order to dissipate the energy that would have been dissipated afterwards if more modes or scales had been simulated.

### 3.2. Resolution of Burguers equation

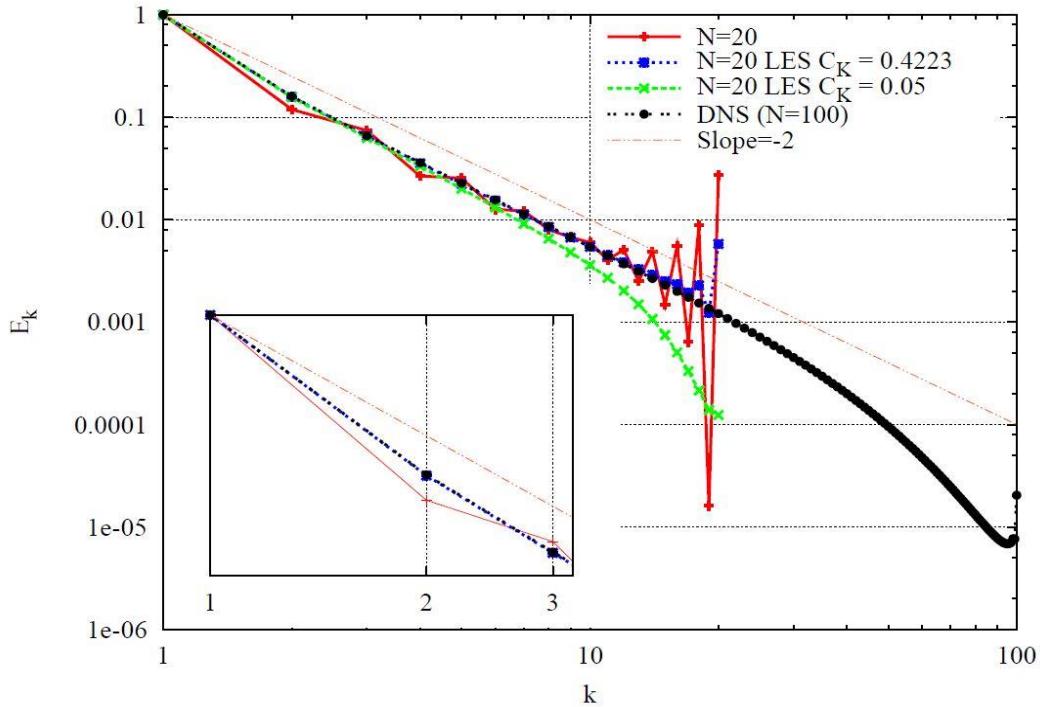


Figure 68: The Burguers equation. Reference solution for the energy spectrum of the steady-state solution of the Burguers equation with  $Re = 40$  (extracted from [16])

The Figure 68 shows the reference results of the energy spectrum of the Burgers equation for  $Re = 40$  for a DNS simulation with 20 and 100 Fourier modes and for a LES simulation with 20 Fourier modes but different Kolmogorov constants. Note that  $C_K$  should be 0.4523 and not 0.4223.

When solving the Burgers equation numerically, it is important to notice that an equation for each mode can be solved separately. Then, the most important point to compute is the convective term seen in equation (159): the code works with complex numbers, taking the real part and the imaginary part of the velocity  $\hat{u}_k$  (see Attachment 1 for the code). After computing the convective term the new value of the velocity can be computed and so on until the solution has converged.

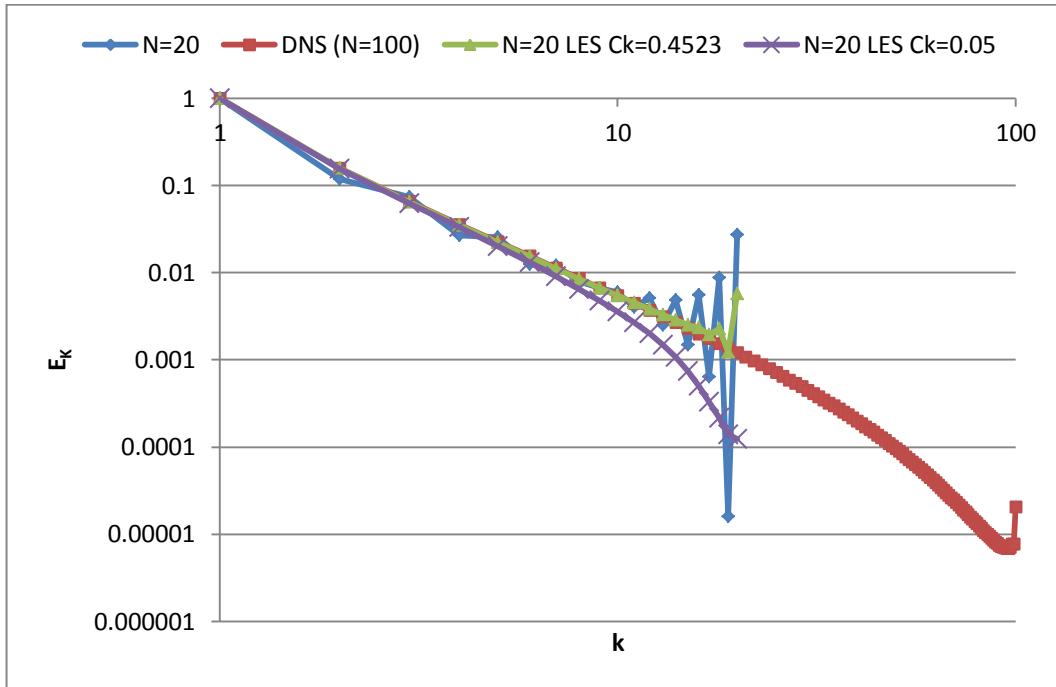
In the DNS simulations, the viscosity is equal to  $Re^{-1}$ , but in the LES simulations, the viscosity is redefined using equation (161) adding the extra viscosity in order to dissipate more:

$$\nu(k) = \frac{1}{Re} + \nu_t^{+\infty} \left( \frac{E_{k_N}}{k_N} \right)^{\frac{1}{2}} \nu_t^* \left( \frac{k}{k_N} \right) \quad (165)$$

So the diffusive term is redefined as:

$$-k^2 \nu(k) \hat{u}_k \quad (166)$$

The *Figure 69* shows the results obtained with the simulation of the 1D Burgers equation for  $Re = 40$ . In order to see the numerical values of the simulation, please refer to *Attachment 2*.



**Figure 69: The Burgers equation. Results obtained with the simulation of the Burgers equation for  $Re = 40$**

As it can be seen, both the reference results and the results obtained with the simulation are similar. From these results, it is possible to extract some conclusions about a turbulence simulation.

First of all, with a DNS (Direct Numerical Simulation) it can be seen that in order to obtain good results, a large amount of modes or scales need to be simulated (up to 100 in this case). When doing a DNS with only 20 modes, the energy is not well distributed as it happens with the DNS with 100 modes and the energy backscattering produces instabilities in all the scales, giving bad results even in the easiest zone to be solved (first modes).

When studying the resolution with a LES model (Large-Eddy Simulation) the number of simulated modes is only 20, but with the Kolmogorov constant taking the value 0.4523 the results obtained for the large scales are quite similar to the results obtained with the DNS of 100 modes and only some instabilities appear on the last modes. Another interesting result is the LES simulation with the Kolmogorov constant equal to 0.05. In this case, there is higher energy dissipation when increasing the mode and as it can be seen in *Figure 69*, the energy spectrum decays quicker than in the other simulations.

Both cases are solved in a short term of time, but it has been noticed that the LES model is solved quicker than the DNS with 100 modes, making it more interesting as a resolution method for turbulence.

## 4. CONCLUSIONS

This study was proposed by the *Centre Tecnològic de Transferència de Calor* from the Universitat Politècnica de Catalunya in order to introduce the student into the field of the computational fluid dynamics. The CTTC offered some seminars where the theory behind the discretization of the governing equations was provided to the student and, afterwards, a tutorship was offered to the student during the self-working of the validation problems to test the own developed code.

The study has shown that the developed code, according to the theory given in these pages, can solve different testing problems such as the conduction problem proposed by CTTC, the Smith-Hutton case, the Driven Cavity case and the Differentially Heated Cavity test. As a particular application, the code has been used to solve the flow around a square cylinder inside a plane channel increasing the Reynolds number and the flow around two square cylinders whose separation is increased gradually in order to study how the interferences between them change. These last two exercises have a strong relationship with the field of the aerodynamics, which is one of the main subjects of the aerospace studies, since some aerodynamic parameters such as the drag coefficient and the lift coefficient have been computed.

In general, the code could solve all the testing problems, but when the velocity gradients are too large, or the case is turbulent and there is a non-steady situation, the obtained results start to differ from the references. This happens because the code is only prepared for steady and laminar problems. The only way to improve the results would be to implement other types of more accurate discretization or increasing the mesh refinement, which yields to a not recommended increase of the computational costs. With the need of improving the results for turbulent flows, an introduction to turbulence and different ways to solve it has been done in order to see the advantages of these different methods and from where they come.

### 4.1. Improvements and future work

The developed code has fulfilled the main goals of this study as it has been shown, but still some improvements and further work can be done related to this study, which could yield to another project about turbulence and its resolution.

As improvements or future work related to this study, the following points are proposed:

- Depuration and improvement of the efficiency of the code, especially when working with non-equidistant meshes (need to check if the code is really well implemented).
- Resolution of the problems proposed in [10] and [11] (square cylinder) with a finer mesh in order to obtain better results.
- Resolution and validation of the results of the interferences generated by an array of square cylinders proposed in [11].
- Generation of a code prepared to solve turbulent cases and improve the results with coarser meshes.



## 5. SOFTWARE

The software used to develop this study has been the *Microsoft Office 2010*, particularly the *Word* and *Excel* utilities, the *Dev-C++ 4.9.9.2* from *Bloodshed Software*, the *Matlab R2012a* from *MathWorks*, *PlotDigitizer 2.6.6* from *Sourceforge* and *Mendeley Desktop 1.14* from *Mendeley Ltd.*.

<b>Microsoft Office 2010</b>	
- Microsoft Word	Elaboration of the documents.
- Microsoft Excel	Data post-processing.
<b>Dev-C++ 4.9.9.2</b>	Elaboration of the codes and generation of the executable files.
<b>Matlab R2012a</b>	Data post-processing.
<b>PlotDigitizer 2.6.6</b>	Data pre-processing.
<b>Mendeley Desktop 1.14</b>	References manager.

### 5.1. Licenses

<b>Microsoft Office 2010</b>	Students license obtained for 79.00 €.
<b>Dev-C++ 4.9.9.2</b>	Free software obtained from <a href="http://www.bloodshed.net">www.bloodshed.net</a>
<b>Matlab R2012a</b>	Home license obtained for 105.00 €. This option has been chosen since the program is only used for data post-processing and there is no planned commercial distribution of the study.
<b>PlotDigitizer 2.6.6</b>	Free software obtained from <a href="http://www.plotdigitizer.sourceforge.net">www.plotdigitizer.sourceforge.net</a>
<b>Mendeley Desktop 1.14</b>	Free software, but the license for the premium account has been obtained through the Universitat Politècnica de Catalunya without any cost.

All the licenses costs have to be included in the budget of the study.



## 6. TASKS PLANNIFICATION

Finally, the calendar proposed in the project charter of this study has been modified and it was not possible to follow the original plans. Here, a comparison and an update of the calendar are done.

### 6.1. Tasks

**Task 1:** Resolution of the Navier – Stokes equations (and turbulence if applicable) and validation and verification of the code (includes the elaboration of the theory in the report and the elaboration of the codes).

**Task 2:** Identification and measurement of the convenience of solving the Navier – Stokes equations in the field of the selected engineering problem (includes the preparation of the developed code to solve the selected engineering problem, which has been the resolution of the flow around a square cylinder and the generation of interferences between two square cylinders).

**Task 3:** Evaluation and conclusions of the results obtained in the simulation of the selected engineering problem.

### 6.2. Dependencies among tasks

TASK	ESTIMATED TIME	REAL TIME	PARALLELISM	DEPENDENCIES
1	5 months	6.5 months	2,3	-
2	2 months	2 months	1,3	1,3
3	3 months	3 months	1,2	1,2

Table 23: Dependencies among tasks

### 6.3. Gantt chart

	February	March	April	May	June
Task 1					
Task 2					
Task 3					

Table 24: Initial Gantt chart

	February	March	April	May	June	July	August
Task 1							
Task 2							
Task 3							

Table 25: Final Gantt chart

### 6.4. Future tasks

Here, an estimation of the necessary time to develop the future work proposed in 4.1 is shown.

- Depuration and improvement of the efficiency of the code: **2 to 3 weeks**.
- Resolution of the problems proposed in [10] and [11] (square cylinder) with a finer mesh in order to obtain better results: **2 to 3 weeks**.

- Resolution and validation of the results of the interferences generated by an array of square cylinders proposed in [11]: **3 weeks**.
- Generation of a code prepared to solve turbulent cases and improve the results with coarser meshes: **6 to 8 weeks**.

These periods of time are set separately and no parallel work is supposed.

## 7. ENVIRONMENTAL IMPACT

This study has been developed mainly with the use of a computer and there is no final product which can yield to a direct environmental impact with its production. However, the following points have to be taken into account:

- **Paper consumption:** printing of studied documents and papers used to write down ideas and to take notes. As last point, printing of the report and further documents related with this study.
- **Electric power:** while using the computer and running the simulations a lot of energy was consumed, also in order to make the computer run faster, the processor was switched to high performance mode, increasing the consumption. This is the activity with more environmental impact done during this study.

The environmental impact of these previous activities is hard to measure since they are present in the everyday life of any person and especially of a student. Therefore, and in order to minimize the environmental problems that this study may cause, the following measures have been taken:

- **Reduction of the used paper:** use of recycled paper, use of all the available space in the paper, printing using both sides of the paper when necessary, printing the report and all the documents once it is sure that everything is correct, etc.
- **Reduction of the electric power consumption:** activation of the high performance mode of the processor only when running simulations (while developing the documents or doing the data post-processing, the processor was switched to battery saving mode or economical mode). The high performance mode was configured to shut down the screen of the computer after one minute of inactivity, making the computer only consume energy due to the simulations.



## 8. REFERENCES

- [1] CTTC - Universitat Politècnica de Catalunya, "NUMERICAL SOLUTION OF CONVECTION," pp. 1–15, 2014.
- [2] S. V. Patankar, *Numerical heat transfer and fluid flow*. Hemisphere Publishing Corporation, 1980.
- [3] CTTC - Universitat Politècnica de Catalunya, "Introduction to the Fractional Step Method," pp. 1–10, 2014.
- [4] E. Simons, "An efficient multi-domain approach to large eddy simulation of incompressible turbulent flows in complex geometries," 2000.
- [5] "<http://www.nacad.ufrj.br/~rnelias/gallery/cavity.html>." .
- [6] CTTC - Universitat Politècnica de Catalunya, "Driven Cavity (Abstract)." 2014.
- [7] G. De Vahl Davis, "Natural convection of air in a square cavity: a bench mark numerical solution," *Int. J. Numer. Methods Fluids*, vol. 3, no. July 1982, pp. 249–264, 1983.
- [8] G. De Vahl Davis and I. P. Jones, "Natural convection in a square cavity: a comparison exercise," *Int. J. Numer. Methods Fluids*, vol. 3, no. July 1982, pp. 227–248, 1983.
- [9] "[https://en.wikipedia.org/wiki/Simpson%27s\\_rule](https://en.wikipedia.org/wiki/Simpson%27s_rule)." .
- [10] M. Breuer, J. Bernsdorff, T. Zeiser, and F. Durst, "Accurate computations of the laminar flow past a square cylinder based on two different methods: Lattice-Boltzmann and finite-volume," *Int. J. Heat Fluid Flow*, vol. 21, no. 2, pp. 186–196, 2000.
- [11] Y. Bao, Q. Wu, and D. Zhou, "Numerical investigation of flow around an inline square cylinder array with different spacing ratios," *Comput. Fluids*, vol. 55, pp. 118–131, 2012.
- [12] "[http://www.sfu.ca/~ptaherib/teaching/ENSC\\_283\\_2013/Project/ENSC283\\_Project.pdf](http://www.sfu.ca/~ptaherib/teaching/ENSC_283_2013/Project/ENSC283_Project.pdf)." .
- [13] "[https://en.wikipedia.org/wiki/Navier%20%93Stokes\\_equations](https://en.wikipedia.org/wiki/Navier%20%93Stokes_equations)." .
- [14] "<http://www.gaussianwaves.com/2014/07/how-to-plot-fft-using-matlab-fft-of-basic-signals-sine-and-cosine-waves/>." .
- [15] U. Frisch, "Turbulence. The Legacy of A.N.Kolmogorov," *Cambridge Univ. Press*, 1995.
- [16] CTTC - Universitat Politècnica de Catalunya, "Burgers equation in Fourier space," vol. 4, no. 4, pp. 1–8, 2014.
- [17] J. Smagorinsky, "General Circulation Experiments with the Primitive Equations," *J. Fluid Mech.*, vol. 91:99–164, 1963.
- [18] M. Lesieur and O. Metais, "Spectral large-eddy simulation of isotropic and stably stratified turbulence," *J. Fluid Mech.*, vol. 239:157–19, 1992.

\*The documents from the CTTC-Universitat Politècnica de Catalunya are found, as free-share content, in the following link: <https://www.cttc.upc.edu/public/courses/>