POLITECNICO DI TORINO

Corso di Laurea Magistrale in Ingegneria Energetica e Nucleare

Tesi di Laurea Magistrale

NUMERICAL SIMULATION OF HEAT AND MASS TRANSFER PHENOMENA APPLIED TO FLAT-PLATE SOLAR COLLECTORS

Relatore:
prof. Lanzini Andrea

Relatore estero:
prof. Kizildag Deniz

Candidato:
Edoardo Nervo

Anno Accademico 2019/2020

Summary

Numerical methods and simulations in fluid dynamics and Heat Transfer (HT) are constantly growing in terms of number of both courses offered at universities and active researches in the field. Nowadays, Computational Fluid Dynamics (CFD) codes are being consolidated as design tools by the industry and both commercial and open-source codes are available. Anyway, users of CFD need to have a fully understanding of the numerical methods implemented in these more complex software in order to give meaning and accuracy to their results.

Therefore, the two main objectives of this project are: firstly, to acquire a basic training in the numerical resolution of the governing equations in heat transfer and fluid dynamics problems, acquiring the skills to critically judge their quality through validation techniques and, secondly, apply the developed knowledge to a real-world case like the heat transfer study of a flat-plate solar collector.

The project is organized like a work-flow that starts from the most basic numerical methods and problems to the most complex ones; finalizing the work with a study of a real flat-plate collector. Chapter 1. *'Introduction'* contains the work-flow description and a brief presentation of all the case-study analyzed. Chapter 2. *'Discretization methods and solvers'* contains the fundamental numerical methods with which physical phenomena, described through appropriate differential equations, are analyzed. Chapter 2. *'Heat conduction methods'* is the construction base of all the methods implemented in this work, therefore emphasis on concepts and calculation details are given here. Chapter 4. *'Convection and diffusion'* is focused on the resolution of convection and diffusion problems, with the flow field known in advance. The Smith-Hutton problem is addressed in this chapter since many of the features commonly encountered in practical convection-diffusion problems. Different numerical schemes are presented in and their pros and cons are described. The calculation of the velocity field itself is finally treated in Chapter 5. *'Incompressible flow method using the Navier-Stokes equations'*. This chapter describes the implementation of the Fractional Step Method (FSM) in the solution of the Navier-Stokes equations with the aim to obtain solutions for two famous benchmark problems: the Lid-Driven Cavity problem and the Differentially Heated Cavity problem (DHC). In the DHC problem, momentum equations are coupled with the energy equation. The problems presented and solved are intended to be a material base over the analysis of flat-plate solar collector in Chapter 6. *'Numerical analysis of a flat-plate solar collector'*. In the last chapter an analysis of a typical flat-plate solar collector is performed, with a simple one-dimensional (1D) model. The heat transfer between the absorber plate and the cover is then investigated deeply and the results are compared to the ones obtained with the 1D model.

Acknowledgments

Throughout the writing of this project I have received a great deal of support and assistance.

I would first like to thank my supervisor, Professor Deniz Kizildag, who provided me with the project guidelines and material, being always concerned, despite the distance imposed by the pandemic, that the work would continue well and smoothly. In the same way I would like to thank also Àdel Alsalti, who helped me several times with the most practical aspects of the project, always available, competent and professional.

Thanks to all the friends who have been close to me in this project, both in Barcelona and in Italy. In particular, to the guys in my study group Giorgio, Federico, Leonardo and Giulia and to my colleague Francesco who gave me a little help for the last part of the project.

Finally, I would like to thank my parents for their wise counsel and support for all my ideas, projects and madness. Without you none of this would have been possible.

Contents

List of Figures

List of Tables

1. Introduction

This work is intended to provide knowledge in computational and numerical simulation by means of the development, validation and verification of HT and CFD codes, which will be related to the resolution of the conservation equations of mass, momentum and energy applied firstly to several benchmark problems and secondly to a specific case on a field of interest.

Heat transfer phenomena plays an important role in many industrial and environmental problems. There is not a single application in the industrial engineering that does not involve, in some way, HT effects and their accurate prediction. It is widely involved in the generation of power from conventional fossil fuels, nuclear sources and many renewable sources too. Heat transfer processes determine the design of industrial equipment such as boilers, condensers, turbines and many more. Quite often the challenge is to maximize heat transfer rate (such as in heat exchangers), or to minimize it (like in insulation-related studies). HT plays an important part in the design of solar energy conversion systems for water and space heating, cooling of electronic equipment, or refrigeration and air conditioning systems. Finally, HT issues also occur in air and water pollution problems and strongly influence climate at local and global scale.

Computational fluid dynamics is a branch of fluid mechanics that uses numerical analysis and data structures to analyze and solve problems that involve fluid flows. In the last years CFD has experiences rapid advances and the potential it provides to analyze complex thermal systems is of considerable interest to the energy engineer. CFD is applied to a wide range of research and engineering problems in many fields of study and industries, including aerodynamics and aerospace analysis, weather simulation, natural science and environmental engineering, industrial system design and analysis, biological engineering and engine and combustion analysis.

Throughout this work, consolidated numerical methods for HT and CFD are presented mathematically and are verified through the study of benchmark cases. The work-flow is organised with the following logic: the numerical methods are presented first for the heat transfer part and then for the CFD part. The models are presented in order of complexity and are always described mathematically, implemented in a code and verified on benchmark cases. This presentation process starts with the one-dimensional heat transfer analysis until the study of the natural convention in a rectangular cavity inclined at a defined angle. Finally, knowledge of these methods is applied to a practical case of heat transfer in a solar collector, demonstrating a real application of these models.

1.1. Objectives

The objectives of the project can be summarized as:

- Acquire a basic training in the numerical resolution of the governing equations in the fluid dynamics and heat and mass transfer;
- Acquire experience in the field of programming and verification of HT and CFD codes, along with a correct data visualization;
- Learn the ability to critically judge a general CFD HT code;
- Acquire knowledge of the models commonly used to describe a flat-plate solar collector;
- Implement acquired knowledge of CFD HT phenomena to a real case-study in a flat-plate solar collector.

1.2. Scope

- Discretization methods and solvers:
  General mathematical approach of the problems involved in the integration of the equations of fluid dynamics and heat and mass transfer. The finite difference method, the finite volume method, and the presentation of solvers are presented in this chapter. The finite volume method is favored in this text;
- Heat conduction methods:
  Development of the methodology used to solve the heat equation, based on finite volume techniques applied to orthogonal, uniform and structured meshes. The resolution of the linear systems of the discretization equations is done with direct and iterative methods i.e. Gauss-Seidel method (GS), Tri-Diagonal Matrix Algorithm (TDMA) and LU factorization. Moreover, a two-dimensional (2D) conduction HT benchmark problem in transient regime is presented, solved and validated;
- Convection and diffusion:
  The generic form of the transport equations with the convective terms is presented. The different techniques of integration of the equation and the problems of precision (false diffusion) and convergence (stability) that may occur are explained. Furthermore, a 2D convection-diffusion benchmark problem (the Smith-Hutton problem) is presented, solved and validated;
- Incompressible flow method using the Navier-Stokes equations:
  The problem of solving the Navier Stokes (NS) equations is presented, both from a physical and numerical point of view. The methodology explained is based on the Fractional-step method. Additionally, two 2D fluid dynamics benchmark problems, the Driven cavity flow problem and the Differentially Heated Cavity are presented, solved, validated and verified;

- Numerical analysis of a flat-plate solar collector:
  An introductory part on the functioning of a flat-plate solar collector (FPSC) is presented, followed by a description of a one-dimensional numerical model typically used to study their performance. One aspect of this model, the heat transfer between the absorber plate and the cover, is then deepened and analysed with a two-dimensional model, in order to apply the acquired knowledge and make a comparison between the 2D and 1D model.

2. Discretization methods and solvers

2.1. Finite volume method

The finite volume method (FVM) is a numerical method used to solve partial differential equations (PDE) in the form of algebraic equations, using conservation laws [1]. The FVM evaluates a generic variable at discrete locations in the computational domain. This method divides a domain with a generic geometry into a finite number of elements (a mesh), successively used to build finite control volumes (CV). The discretization of the domain can be performed by implementing a vertex-centred approach (where the nodes of the mesh are the centres of the finite volume and the boundaries are obtained by connecting the centroids of each element), or a cell-centred approach (where control volumes coincide with elements) as shown in Figure 1. After the domain composition, an integral formulation of the balance

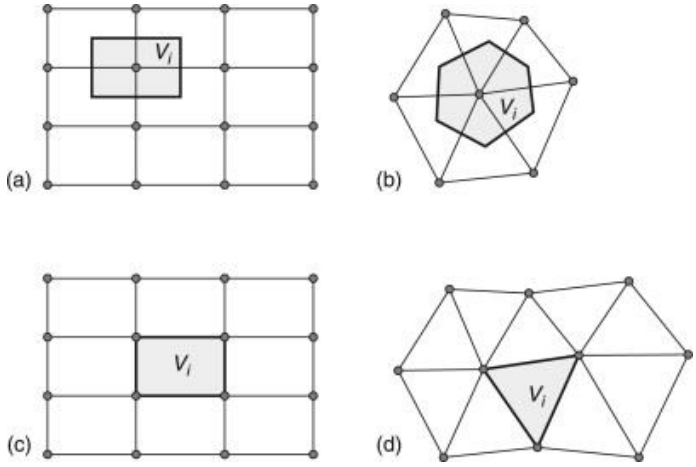


Figure 1: Mesh and control volumes in vertex-centred FVM (a,b) and cell-centred FVM (c,d). Control volumes are highlighted by grey-coloured [1]

equations and a successive approximation of integrals by numerical integration are needed for each control volume [1]. Considering, for example, the mass transport balance for an incompressible fluid, under the assumptions of stationary mesh and neglecting the source

terms, the conservation law can be written:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot \rho v = 0 \tag{2.1.1}$$

where ρ is the density and v is the velocity vector. The FVM enforces equation 4.1.1 for a small control volume defined by the computational mesh so that, for the ith cell, defined by a volume V_i and a boundary surface S_i:

$$\int_{V_i} \frac{\partial \rho}{\partial t} dV + \int_{V_i} \nabla \cdot \rho v \, dV = 0 \tag{2.1.2}$$

Since finite volume methods discretize the balance equation directly, an obvious virtue of such methods is the conservation property (the flux entering a given volume is identical to that leaving the adjacent volume). Because of this feature, the FVM has proved to be very suitable for the solution of problems in fluid mechanics, as well as in heat and mass transfer. Equation 2.1.2 can be rewritten by volume-averaging the first term and applying the Gauss theorem to the second term. Simple manipulations yield:

$$\frac{\mathrm{d}\rho_i}{\mathrm{d}t} + \frac{1}{V_i} \oint_{S_i} \rho v \cdot n \, dS = 0 \tag{2.1.3}$$

where n is the normal vector to S_i, outward from V_i. Function values and derivatives can be now approximated in a finite difference (FD) fashion, the details are discussed in the further sections. The equations written for each volume are then assembled, providing a unique algebraic system to be solved using numerical methods.

2.2. Discretization methods

A numerical solution is a set of numbers from which the distribution of the dependant variable can be constructed[2]. It should be clear that only a finite number of numerical values can be computed, although this number can be made large enough to satisfy the practical purposes for certain problems. Considering a generic dependant variable ϕ, a discretization equation is an algebraic relation connecting the values of ϕ for a group of grid points. The exact solution of the differential equation is replaced by the solution of the discretized equation which focuses the attention on the values at the grid points, passing from a continuous domain to a discretized one composed by a finite number of points. For a given differential equation, the discretization equation can be derived in many ways. The usual procedure for deriving the FD equations consists of approximating the derivatives with truncated Taylor series.

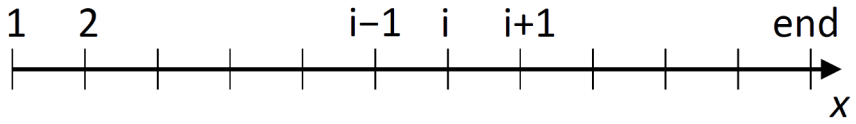


Figure 2: 1D equally-spaced grid

Considering Figure 2 , where a 1D grid equally spaced is presented, it can be derived from the Taylor expansion that:

$$\left(\frac{\mathrm{d}\phi}{\mathrm{d}x}\right)_i = \frac{\phi_{i+1} - \phi_{i-1}}{2\Delta x} \tag{2.2.1}$$

$$\left(\frac{\mathrm{d}^2\phi}{\mathrm{d}x^2}\right)_i = \frac{\phi_{i+1} - \phi_{i-1}}{\Delta x^2} \tag{2.2.2}$$

where Δx is the distance between two adjacent grid point. The substitution of these expressions into the differential equation leads to a FD equation.

2.3. Explicit, Crank-Nicolson and Implicit schemes

Equations 2.2.1 and 2.2.2 are used to discretize space derivatives, while regarding the time derivative other schemes have to be introduced. In order to discuss the discretization in time, it is convenient to introduce a weighting factor θ, so that a generic derivative $d\Phi/dt$ can be discretized as:

$$\frac{\Phi^{n+1} + \Phi^n}{\Delta t} = (1-\theta)\Phi^n + \theta\Phi^{n+1} \tag{2.3.1}$$

where n denotes the "old" (given) values of the function at the grid points at time t and $n+1$ denotes the "new" (unknown) values at time $t + \Delta t$. For certain specific values of the weighting factor θ, the discretization equation reduces to one of the well-known schemes for parabolic differential equations. In particular, $\theta = 0$ leads to the explicit Forward-Euler (FE) method, $\theta = 0.5$ to the Crank-Nicolson (CN) scheme and $\theta = 1$ to the implicit Backward-Euler (BE) method.

The FE method allows the possibility to calculate the property Φ^{n+1} explicitly from the known values of Φ^n. Thanks to this characteristic, this scheme doesn't require the solution of a set of simultaneous equation but its convenience is, however, offset by a serious limitation due to numerical instabilities. For example, in a heat conduction one-dimensional problem with uniform conductivity and mesh, this condition can be expressed as:

$$\Delta t < \frac{\rho c (\Delta x)^2}{2k} \tag{2.3.2}$$

where c is the specific heat and k is the heat conductivity. If this condition is not respected, physically unrealistic results could emerge. The problem about equation 2.3.2 is that as Δx is reduced in order to increase the spatial accuracy, a use of a smaller Δt is forced (perhaps unnecessarily).

The Crank-Nicolson scheme ($\theta = 0.5$) is usually described as unconditionally stable [2]. The "stability", in mathematical sense, simply ensures that some oscillations will eventually die out but it doesn't ensure physically acceptable solutions, so that, if a too "large" time-step is chosen, it is likely to encounter oscillatory behaviour.

The BE method ($\theta = 1$), finally, satisfies the requirements of simplicity and physically satisfactory behavior. On the other hand, the BE scheme requires the solution of a set of linear equations and it arises the need of the implementation of a solver. It must be admitted, however, that for small time-step the CN scheme is more accurate then the BE scheme.

2.4. Boundary conditions

Each CV is defined by one algebraic equation. Volume integrals are calculated in the same way for every CV, but fluxes through CV faces coinciding with the domain boundary require special treatment. These boundary fluxes must either be known, or be expressed as a combination of interior values and boundary data. Since they do not give additional equations, they should not introduce additional unknowns. Since there are no nodes outside the boundary, these approximations must be based on one-sided differences or extrapolations. The different typologies will be discussed in details in the next sections, directly applied at the different problems presented.

2.5. Solution of Linear Equation Systems

It should be noted that, while constructing the discretization equations, we mold them into a linear form but do not assume a specific method would be used for their solution. Therefore, any suitable solution method can be employed at this stage. It is useful to consider the derivation of the equations and their solution as two distinct operations, and there is no need that one influences the other.

2.5.1. Tri-diagonal matrix algorithm

The solution of a set of discretized equations for the one-dimensional situation can be obtained with the Gauss-elimination method. Due to the particular simplicity of the set of equations in the one-dimensional case, the elimination process turns into a convenient algorithm, called the Thomas algorithm or TDMA (TriDiagonal-Matrix Algorithm). TDMA refers to the fact that all the non-zero elements of the coefficients matrix align themselves along three diagonals of the matrix.

The detailed presentation of TDMA will not be presented here because it is not directly the scope of this book and the literature is full of detailed and rigorous explanation of the algorithm [2].

This algorithm is a very robust and efficient direct solver: unlike the more general matrix methods, the TDMA requires computer storage and computer time proportional only to N, rather than N^2 or N^3.

2.5.2. Gauss-Seidel method

The following discussion, as before, aims to explain the general procedure and effectiveness of the algorithm rather than going into the mathematical details, which the literature is plenty of.

One of the simplest iterative methods is the Gauss-Seidel method in which the values of the variable (the temperature in this example) are calculated by visiting each grid point in a certain order. An initial guess on the temperature field is required in order to start the algorithm in the beginning, then, for the following steps, the values from the previous operation will be used. As each grid point is visited, the corresponding value of temperature is calculated from the neighbors grid points temperature values. If the discretization equation can be written as:

$$a_P T_P = \sum a_{nb} T_{nb} + b \tag{2.5.1}$$

where the subscript nb represents for the neighbor-point value in the computer storage, the the grid point temperature T_P can ba calculated as:

$$T_P = \frac{\sum a_{nb} T^*_{nb} + b}{a_P} \tag{2.5.2}$$

where T^*_{nb} is the neighbor point value present in the computer storage. When all grid points have been visited, one iteration of the algorithm is completed.

The Gauss-Siedel method does not always converge, indeed, a criterion exits in order to guarantee the convergence of the method. The criterion was formulated by Scarborough [2] illustrating a sufficient condition for the convergence of the Gauss-Siedel method, that is:

$$\frac{\sum |a_{nb}|}{|a_P|} \leq 1 \tag{2.5.3}$$

for all equations, and

$$\frac{\sum |a_{nb}|}{|a_P|} < 1$$

for at least one equation.

The iteration is terminated when a prescribed convergence criterion is satisfied, such as:

$$|T^k_P - T^{k-1}_P| \leq \varepsilon \tag{2.5.4}$$

where ε is an error in the grid point temperature considered acceptable and k refers to the level of the iteration.

Although the convergence of the Gauss-Siedel method can be granted, one of the major disadvantage is that its converge could be too slow when a "large" number of grid points are involved. The reason of this slowness is due to the fact that the method transmits the boundary condition information at a rate of one grid interval per iteration.

2.5.3. LU factorization

Let $A \in \mathbb{R}^{n\times n}$. Two suitable matrices L and U, lower triangular and upper triangular, can be constructed such that:

$$A = LU \tag{2.5.5}$$

Equation 2.5.5 is an LU-factorization of the matrix A. It can be demonstrated that, if A is non-singular, so both L and U and thus their diagonal elements are non-null.[3] In such a case, solving **Ax = b** leads to the solution of the two triangular systems:

$$Ly = b, \quad Ux = y \tag{2.5.6}$$

Both systems are easy to solve with a *forward substitutions algorithm* and a *backward substitutions algorithm*. Details can be found in [3]. Determining the elements of the factors L and U requires about $2n^3/3$ operations.[3]

2.5.4. Direct and Iterative methods

The reader will notice that, when moving to multi-dimensional domains, the discretizations process of the differential equations is a straightforward extension of the one-dimensional case. However, regarding the resolution algorithm, that is not true. Indeed, the simple tridiagonal-matrix algorithm (TDMA) can't be used anymore due to the penta-diagonal nature of the coefficient matrix in the two-dimensional problems.

Direct methods compute the solution to a problem in a finite number of steps. These methods would give the precise answer if they were performed in infinite precision arithmetic. Examples include Gaussian elimination and the LU-factorization method for solving systems of linear equations. [4]

In contrast to direct methods, iterative methods are not expected to terminate in a number of steps. Starting from an initial guess, iterative methods form successive approximations that converge to the exact solution only in the limit of a convergence criterion. Examples include Gauss-Seidel method, Krylov subspace methods and multigrid methods. In computational matrix algebra, iterative methods are generally needed for large problems.

Iterative methods are more common than direct methods in numerical analysis for certain reasons. Direct methods for solving the algebraic equations could be a possibility but, in multi-dimensional problems, they are much more complicated and require large amounts of time and computer storage. A direct method may be acceptable for a linear problem,

due to the fact that the algebraic equations have to be solved only once; but in nonlinear problems, since the single equations have to be solved repeatedly with updated coefficients, direct methods are usually not efficient [2].

The alternative is the iterative methods; methods that starts from a guessed temperature field, for example, and use the algebraic equations in some manner to obtain an improved field. Successive repetitions of the iterative algorithm should lead to a solution that is sufficiently close to the correct one.

It does not exist the "perfect" choice of a method but it really depends on the problem structure and the required accuracy that one wants to reach.

3. Heat conduction methods

3.1. The Heat Diffusion Equation

The major objective in a conduction-based problem is to determine the temperature field in a medium resulting from conditions imposed on its boundaries. In other words, the temperature distribution has to be found, which represents how temperature varies throughout the medium; once this distribution is known, the conduction heat flux in any point of the medium could be computed from Fourier's law. The knowledge of the temperature field is useful for a lot of engineering problems: for example it could be used to study the structural integrity of a solid through determination of thermal stresses, expansions, and deflections. Another popular application is the optimization the thickness of an insulating material or to determine the compatibility of special coatings or adhesives used with the material [2]. The heat diffusion equation can be derived applying the energy conservation on a differential control volume, identifying the energy transfer processes and introducing the appropriate rate equations. The result is a partial differential equation whose solution is the temperature field of the medium, given certain boundary conditions. Considering a medium within which there is no bulk motion (advection) and the temperature distribution $T(x,y,z)$ is expressed in Cartesian coordinates, the following equation can be obtained:

$$\rho c \frac{\partial T}{\partial t} = \frac{\partial}{\partial x}\left(k_x \frac{\partial T}{\partial x}\right) + \frac{\partial}{\partial y}\left(k_y \frac{\partial T}{\partial y}\right) + \frac{\partial}{\partial z}\left(k_z \frac{\partial T}{\partial z}\right) + \dot{q} \tag{3.1.1}$$

where ρ (kg/m^3) is the density of the medium, c $(J/kg/K)$ is the specific heat capacity, k $(W/m/K)$ is the heat conductivity and $\dot{q}$ (W/m^3) is the rate at which energy is generated per unit volume in the medium .

Equation 3.1.1 is the general form, in Cartesian coordinates, of the heat diffusion equation; its solution is the temperature field $T(x,y,z)$ as a function of time. A more general equation

can be written in the following form:

$$\rho c \frac{\partial T}{\partial t} = \nabla \cdot (k \nabla T) + \dot{q} \tag{3.1.2}$$

valid for an-isotropic medium and general geometry.

3.2. One-Dimensional Steady State Conduction

Considering a one-dimensional steady state problem, in Cartesian coordinates, equation 3.1.1 reduces to:

$$\frac{\mathrm{d}}{\mathrm{d}x}\left(k\frac{\mathrm{d}T}{\mathrm{d}x}\right) + \dot{q} = 0 \tag{3.2.1}$$

where x in the independent space variable. To derive the discretization equation, the grid-point cluster shown in Figure 3 is implemented.[2] The attention is focused on the grid point P, which has the grid point E and W as its neighbors, where E represents the east side (positive x direction) and W stands for the west side (negative x direction). For the one-dimensional problem under consideration, a unit thickness is assumed in the y and z directions. Thus, the volume of the CV shown in Figure 3 is $\Delta x \times 1 \times 1$. Integrating equation 3.2.1 over the control volume:

$$\left(k\frac{\mathrm{d}T}{\mathrm{d}x}\right)_e - \left(k\frac{\mathrm{d}T}{\mathrm{d}x}\right)_w + \int_w^e \dot{q}\, dx = 0 \tag{3.2.2}$$

If the derivatives dT/dx are evaluated with Equations 2.2.1 and 2.2.2 and assuming a

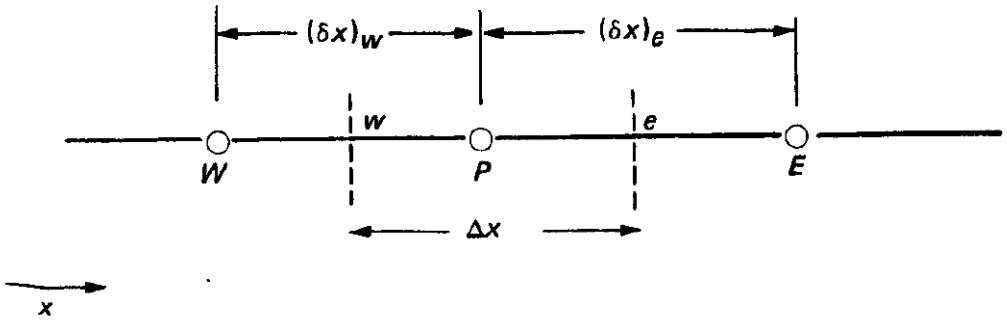


Figure 3: Grid-Point cluster for the one dimensional problem [2]

piecewise-linear profile, the resulting equation will be:

$$\frac{k_e(T_E - T_P)}{(\delta x)_e} - \frac{k_w(T_P - T_W)}{(\delta x)_w} + \dot{q}\Delta x = 0 \tag{3.2.3}$$

where, in this equation, $\dot{q}$ is assumed to be an average value over the control volume. It is useful to rearrange the discretization equation in the following form:

$$-a_W T_W + a_P T_P - a_E T_E = b \tag{3.2.4}$$

where:

$$a_W = \frac{k_w}{(\delta x)_w},$$

$$a_E = \frac{k_e}{(\delta x)_e},$$

$$a_P = a_E + a_W,$$

$$b = \dot{q}\Delta x$$

For the grid points shown in Figure 3, it should be noted that the distances $(\delta x)_e$ and $(\delta x)_w$ does not have to be equal. Indeed, the use of a nonuniform grid is often preferable, allowing a more effectively use of computing power; there is no need to employ a fine grid in regions of the domain where the dependant variable T changes relatively slowly with x. The suitable grid for a problem is not known before the actual computation, however, one normally has the some qualitative expectations, based on physics assumptions, from which some guidance can be obtained [2]. Another way is firstly use a coarse grid in order to find the pattern of $T(x)$ and, secondly, construct a nonuniform grid based on the knowledge acquired.

3.3. Boundary conditions implementation

The heat diffusion equation for the 1D steady state is a second order differential equation, meaning that, in order to find the unique solution, additional boundary conditions equations must be written. Equation 3.2.4 can only be applied to internal nodes of the grid, while there is a need to write two additional equations to treat the nodes at the borders. The same 1D grid as Figure 2 is considered. Usually three kinds of boundary conditions are encountered in heat conduction:

1. Given boundary temperature
2. Given boundary heat flux
3. Boundary heat flux defined through a heat transfer coefficient and the surrounding fluid's temperature

3.3.1. Dirichlet Boundary Conditions

The first boundary condition is straightforward and no additional equations are required. For example, if T_0 is the imposed temperature on the right-side of the grid, it will be sufficient to put:

$$T_{end} = T_0 \tag{3.3.1}$$

3.3.2. Neumann Boundary Conditions

With the second type of boundary conditions, the following equations can be written, depending if the left or the right side are considered:

$$-k_1 \left(\frac{\mathrm{d}T}{\mathrm{d}x}\right)_{x=x_1} = -q \tag{3.3.2}$$

$$-k_{end} \left(\frac{\mathrm{d}T}{\mathrm{d}x}\right)_{x=x_{end}} = -q \tag{3.3.3}$$

where q (W/m^2) is the heat flux imposed on the boundary, it is considered negative because it is supposed to enter the domain. Equations 3.3.2 and 3.3.3 can be simply discretized in:

$$T_2 - T_1 = \frac{q\Delta x}{k_1} \tag{3.3.4}$$

$$T_{end} - T_{end-1} = \frac{q\Delta x}{k_{end}} \tag{3.3.5}$$

3.3.3. Robin Boundary Condition

The third boundary conditions type is similar to the second one but, in this case, the heat flux is generated by convection with a fluid. Two additional equations can be derived, considering the left or right-side boundary:

$$-k_1 \left(\frac{\mathrm{d}T}{\mathrm{d}x}\right)_{x=x_1} = h(T_f - T_1) \tag{3.3.6}$$

$$-k_{end} \left(\frac{\mathrm{d}T}{\mathrm{d}x}\right)_{x=x_{end}} = h(T_f - T_{end}) \tag{3.3.7}$$

that can be simply discretized in:

$$\left(1 + \frac{h\Delta x}{k_1}\right) T_1 - T_2 = \frac{h\Delta x}{k_1} T_f \tag{3.3.8}$$

$$\left(1 - \frac{h\Delta x}{k_{end}}\right) T_{end} - T_{end-1} = -\frac{h\Delta x}{k_{end}} T_f \tag{3.3.9}$$

where h $(W/m^2/K)$ is the heat transfer coefficient and T_f (K) is the surrounding fluid temperature.

3.4. Interface Conductivity

In Equation 3.2.4, the conductivity k_e has been used to define the value of the conductivity on the control-volume face e, while k_w similarly refers to the interface w. There are often

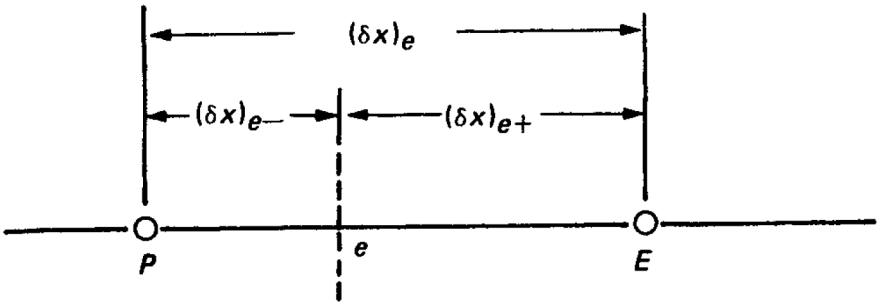


Figure 4: Distances associated with the interface e [2]

cases in which the conductivity depends on the position in the domain; non-uniformity can arise from non-homogeneity of the material, as in a composite slab, or from conductivity variations due to temperature distribution. Referring to Figure 4, the most straightforward method in order to find the interface conductivity k_e is to assume a linear variation of k between grid points P and E [2], so that:

$$k_e = f_e k_P + (1 - f_e) k_E \tag{3.4.1}$$

where f_e is the interpolation factor, defined as the ratio of the distances:

$$f_e \equiv \frac{(\delta x)_{e+}}{(\delta x)_e} \tag{3.4.2}$$

However, it can be shown that this simple-minded approach can lead to rather incorrect results in some cases and it cannot handle the abrupt changes that can occur in a composite material [2]. Considering that, a good representation of the heat flux q_e is one of the main objectives, a steady one-dimensional analysis for the composite slab between P and E can be performed, leading to:

$$q_e = \frac{T_P - T_E}{(\delta x)_{e-}/k_P + (\delta x)_{e+}/k_E} \tag{3.4.3}$$

Then, a combination of Equations 3.4.1 and 3.4.2 yields:

$$k_e = \left(\frac{1 - f_e}{k_P} + \frac{f_e}{k_E} \right)^{-1} \tag{3.4.4}$$

If the interface is placed midway between point P and E, so that $f_e = 0.5$, then:

$$k_e = \frac{2 k_P k_E}{k_P + k_E} \tag{3.4.5}$$

3.5. Unsteady One-Dimensional Conduction

The unsteady one-dimensional heat-conduction equation can be written as:

$$\rho c \frac{\partial T}{\partial t} = \frac{\partial}{\partial x}\left(k \frac{\partial T}{\partial x}\right) + \dot{q} \tag{3.5.1}$$

The discretization techniques presented before can also be extended to the unsteady case. Applying the discretizations at Equation 3.5.1, the unsteady one-dimensional heat conduction equation is:

$$-a_W^{n+1} T_W^{n+1} + a_P^{n+1} T_P^{n+1} - a_E^{n+1} T_E^{n+1} = a_W^n T_W^n + a_P^n T_P^n + a_E^n T_E^n + b^n \tag{3.5.2}$$

where

$$a_W^{n+1} = \frac{\Delta t \theta \alpha_w \Delta x}{(\delta x)_w}$$

$$a_P^{n+1} = 1 + \frac{\Delta t \theta \Delta x}{\rho c}\left(\frac{k_e}{(\delta x)_e} - \frac{k_w}{(\delta x)_w}\right)$$

$$a_E^{n+1} = \frac{\Delta t \theta \alpha_e \Delta x}{(\delta x)_e}$$

$$a_W^n = \Delta t (1-\theta) \frac{\alpha_w \Delta x}{(\delta x)_w}$$

$$a_P^n = 1 - \frac{\Delta t (1-\theta) \Delta x}{\rho c}\left(\frac{k_e}{(\delta x)_e} - \frac{k_w}{(\delta x)_w}\right)$$

$$a_E^n = \Delta t (1-\theta) \frac{\alpha_e \Delta x}{(\delta x)_e}$$

$$b^n = \frac{\dot{q}^n \Delta t}{\rho c}$$

always referring to the grid shown in Figure 3. Moreover, $\alpha = \frac{k}{\rho c}$ (m^2/s) is the thermal diffusivity and $\Delta x = \frac{2}{x_E - x_W}$ is assigned in order to obtain a clearer equation.

3.6. Unsteady Two-Dimensional Conduction

In Figure 5 it is represented the portion of a two-dimensional grid. For the grid point P, points E and W denote its x-direction neighbors while N and S denote the y-direction neighbors. All the nomenclature introduced for the one-dimensional situation can be extended to the two-dimensional situations as well. The PDE relative to the two-dimensional heat equation can be written as:

$$\rho c \frac{\partial T}{\partial t} = \frac{\partial}{\partial x}\left(k \frac{\partial T}{\partial x}\right) + \frac{\partial}{\partial y}\left(k \frac{\partial T}{\partial y}\right) + \dot{q} \tag{3.6.1}$$

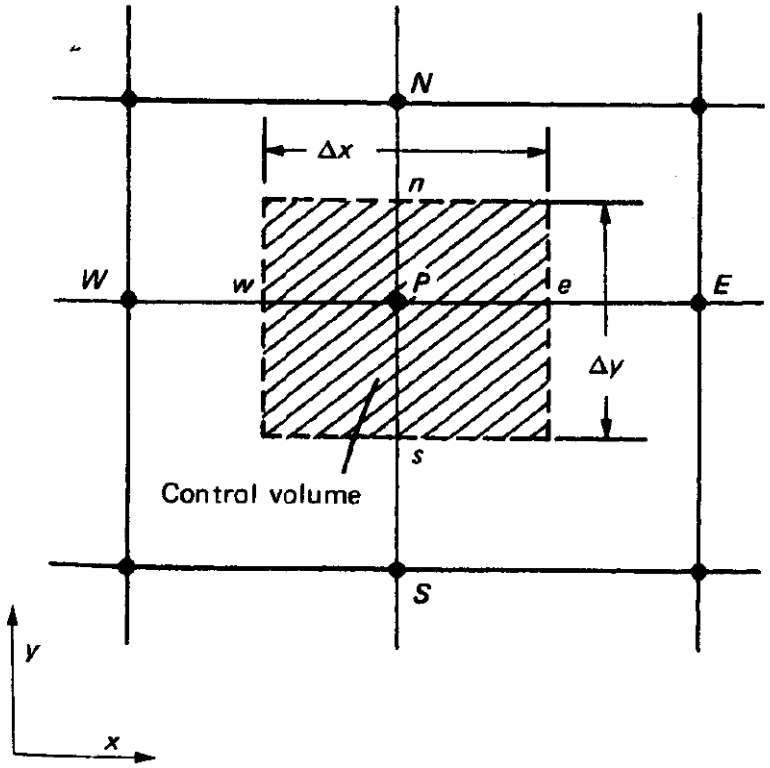


Figure 5: Control volume for the two-dimensional situation

It is straightforward to discretize equation 3.6.1 simply extending the 1D derivations to the two-dimensional case. In this work, the explicit scheme and the implicit scheme are presented. The explicit scheme leads to:

$$T_P^{n+1} = a_W T_W^n + a_E T_E^n + a_P T_P^n + a_N T_N^n + a_S T_S^n + b \tag{3.6.2}$$

where:

$$a_W = \frac{\Delta x \Delta t \alpha_w}{(\delta x)_w}$$

$$a_E = \frac{\Delta x \Delta t \alpha_e}{(\delta x)_e}$$

$$a_P = 1 - \Delta x \Delta t \left(\frac{\alpha_e}{(\delta x)_e} + \frac{\alpha_w}{(\delta x)_w} \right) - \Delta y \Delta t \left(\frac{\alpha_n}{(\delta y)_n} + \frac{\alpha_s}{(\delta y)_s} \right)$$

$$a_N = \frac{\Delta y \Delta t \alpha_n}{(\delta y)_n}$$

$$a_S = \frac{\Delta y \Delta t \alpha_s}{(\delta x)_s}$$

$$b = \frac{\dot{q} \Delta t}{\rho c}$$

Similarly as Δx definition, $\Delta y = \frac{2}{x_N - x_S}$.

Being an explicit method, the grid point temperature T_P depends only on known temperatures so that it does not require the solution of a linear system. However, exactly as the one-dimensional situation, the explicit method is only stable if [5]:

$$\frac{2\alpha\Delta t}{min((\Delta x)^2,(\Delta y)^2)} \leq 1 \tag{3.6.3}$$

having considered a simplified case of a uniform medium and uniform mesh $\Delta x = (\delta x)_e = (\delta x)_w$ and $\Delta y = (\delta y)_n = (\delta y)_s$.

The fully implicit method leads, on the other hand, to the following discretized equation:

$$a_S T_S^{n+1} + a_W T_W^{n+1} + a_P T_P^{n+1} + a_N T_N^{n+1} + a_E T_E^{n+1} = T_P^n + b^n \tag{3.6.4}$$

where:

$$a_S = -\frac{\Delta y \Delta t\, \alpha_s}{(\delta y)_s}$$

$$a_W = -\frac{\Delta x \Delta t\, \alpha_w}{(\delta x)_w}$$

$$a_P = 1 + \Delta x \Delta t \left(\frac{\alpha_e}{(\delta x)_e} + \frac{\alpha_w}{(\delta x)_w}\right) + \Delta y \Delta t \left(\frac{\alpha_n}{(\delta y)_n} + \frac{\alpha_s}{(\delta y)_s}\right)$$

$$a_N = -\frac{\Delta y \Delta t\, \alpha_n}{(\delta y)_n}$$

$$a_E = -\frac{\Delta x \Delta t\, \alpha_e}{(\delta x)_e}$$

$$b = \frac{\dot{q}\Delta t}{\rho c}$$

The coefficients are practically the same as the explicit discretized equation, but its resolution is completely different.

3.7. Boundary conditions implementation

The heat diffusion equation for the two-dimensional case requires boundary conditions similarly to the 1D situation. The same three kinds of boundary conditions will be discussed (given boundary temperature, given boundary heat flux and given boundary heat flux defined through convection with a fluid).

3.7.1. Dirichlet Boundary Conditions

These conditions are again straightforward. If a boundary temperature is imposed, it is sufficient to put all the grid points temperature of the border equal to the known temperature (T_0):

$$T_{edge,i} = T_0 \quad \forall i \tag{3.7.1}$$

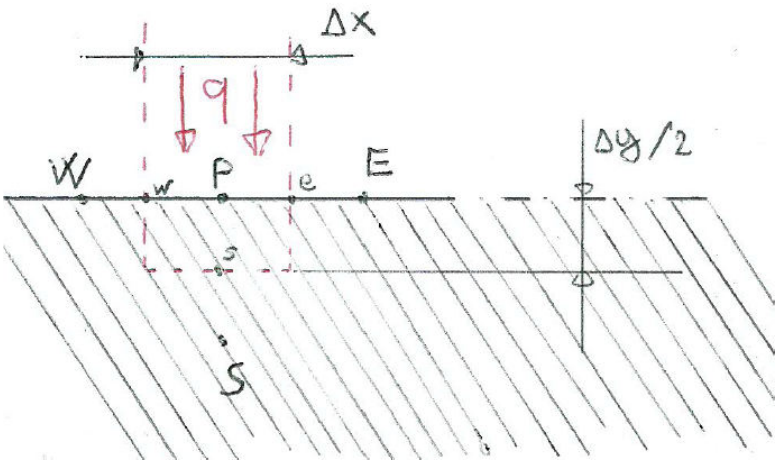


Figure 6: Control volume for an up-side heat flux boundary condition

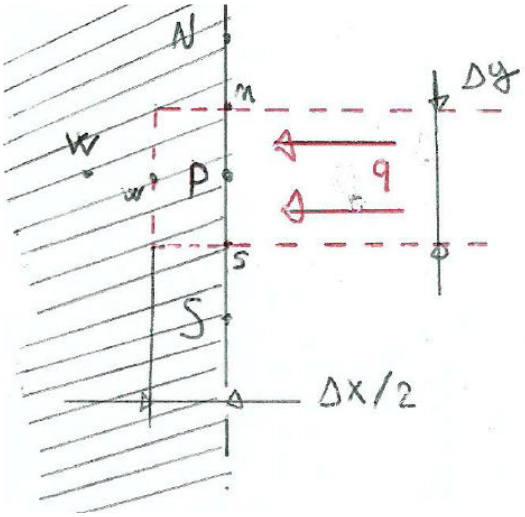


Figure 7: Control volume for a right-side heat flux boundary condition

where i is a generic edge grid point index.

3.7.2. Neumann Boundary Conditions

Two possible examples are shown in Figure 6 and 7.

Differently from the one-dimensional case, the two-dimensional one requires a solution of a set of equations due to the more complex geometry. Applying an energy balance to the case shown in Figure 6, it is possible to write the following equation:

$$a_W T_W + a_P T_P + a_E T_E = b \tag{3.7.2}$$

where:

$$a_W = \frac{k_w \Delta y}{2\Delta x}$$

$$a_P = -\frac{k_s \Delta x}{\Delta y} - \frac{k_e \Delta y}{2\Delta x} - \frac{k_w \Delta y}{2\Delta x}$$

	$x[m]$	$y[m]$
p_1	0.30	0.40
p_2	0.30	0.80
p_3	0.90	0.90

Table 1: Problem coordinates

$$a_E = \frac{k_e \Delta y}{2\Delta x}$$

$$b = -q\Delta x - \frac{k_s \Delta x}{\Delta y} T_S - \frac{\dot{q}\Delta x \Delta y}{2}$$

A similar equation can be written for the right-side situation.

3.7.3. Robin Boundary Conditions

The procedure is similar to the imposed heat flux situation but in this particular case:

$$q = h(T_f - T_P) \tag{3.7.3}$$

where T_f is the external fluid temperature. Deriving the discretized equation for the right-side situation:

$$a_S T_S + a_P T_P + a_N T_N = b \tag{3.7.4}$$

where:

$$a_S = \frac{k_s \Delta x}{2\Delta y}$$

$$a_P = -h\Delta y - \frac{k_w \Delta y}{\Delta x} - \frac{k_n \Delta x}{2\Delta y} - \frac{k_s \Delta x}{2\Delta y}$$

$$a_N = \frac{k_n \Delta x}{2\Delta y}$$

$$b = -hT_f \Delta y - \frac{k_w \Delta y T_W}{\Delta x} - \frac{\dot{q}\Delta x \Delta y}{2}$$

All these set of equations can be solved efficiently with the Thomas Algorithm due to the tridiagonal disposition of the coefficient matrix. In the next section, a two-dimensional transient conduction problem is presented and discussed in the details.

3.8. A Two-dimensional Transient Conduction Problem

3.8.1. Problem definition

A very long rod is composed of four different materials (M1 to M4), represented with different colours in Figure 8. All the lines are parallel to the coordinate axis. The coordinates of the points $p1$ to $p3$ are given in Table 1 and properties of the materials are given in Table 2.

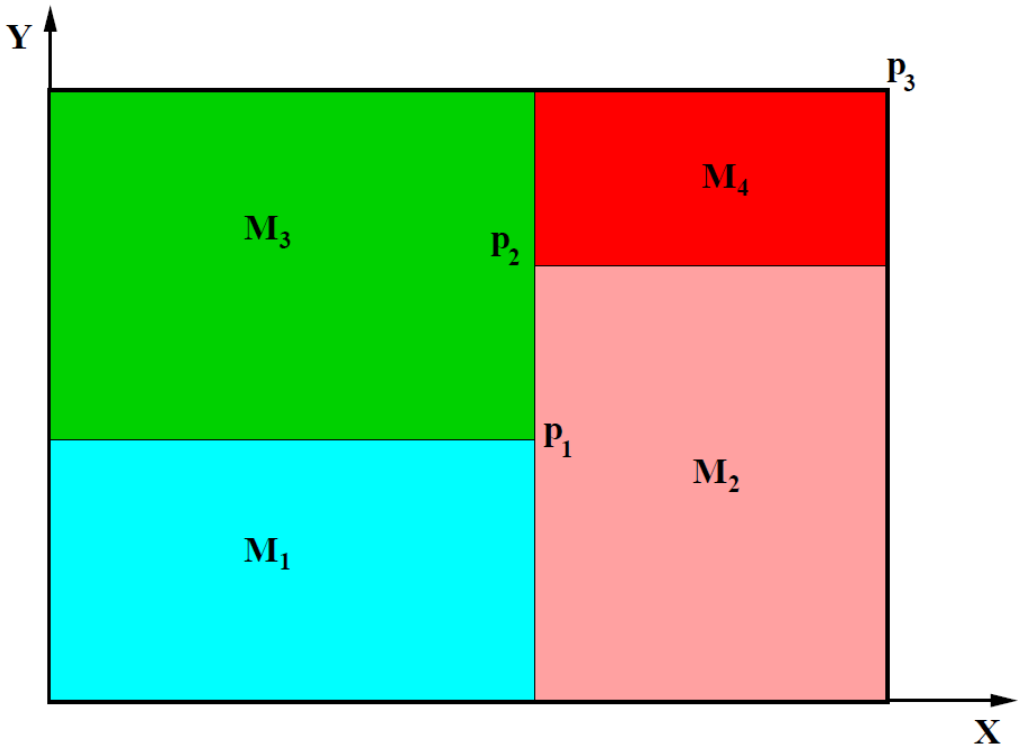


Figure 8: General scheme of the proposed problem

	$\rho[kg/m^3]$	$c_p[J/kgK]$	$k[W/mK]$
M_1	2500	970	180
M_2	2700	930	140
M_3	2200	710	150
M_1	1700	920	140

Table 2: Physical properties

Cavity wall	Boundary condition
Bottom	Isotherm at $T = 18.00\ ^oC$
Top	Uniform heat flow per length $q = 89.00\ W/m$
Left	In contact with a fluid at $T_g = 35.00\ ^oC$ and $h = 8.00\ W/m^2/K$
Right	Uniform temperature $T = 11.00 + 0.006t\ ^oC$ (where t is the time in seconds)

Table 3: Boundary Conditions

Each of the four sides of the rod interacts with the surrounding in a different manner, as described in Table 3. The initial temperature field is $T = 11^oC$. and the transient heat conduction will be studied up to 10000 s.

In order to understand the many differences between a fully explicit and a fully implicit method, both will be implemented in a C++ code and the results will be discussed and compared.

3.8.2. General considerations

The two-dimensional heat diffusion equation, for this case problem is:

$$\rho c \frac{\partial T}{\partial t} = \frac{\partial}{\partial x}\left(k\frac{\partial T}{\partial x}\right) + \frac{\partial}{\partial y}\left(k\frac{\partial T}{\partial y}\right) \tag{3.8.1}$$

Similar to Equation 3.6.1 but without the heat generation term.

The domain considered is a square where, on every side, a different typology of boundary condition is imposed (Dirichlet, Neumann and Robin). Furthermore, the domain is formed by a four different materials with different properties. The composite material complicates the definition of the coefficients of the discretized equations along with the necessary use of the interface conductivity as explained in section 3.4.

The mesh generated for this problem is uniform, for simplicity purposes. Although, the resolution algorithm works with even a non-uniform grid which could lead to similar accuracy with lower computational costs. A non-uniform grid could lead to more precision if it is made with a correct criteria, like refining the grid where the temperature field presents steep gradients.

For both the methods presented, the definition of the coefficients is the same but, obviously, the resolution algorithm is different. The C++ program developed for this purpose aims to be the most reusable as possible, in order to implement the algorithms to successive and different problems.

The domain is discretized in the x-direction and y-direction, obtaining a finite number of control volumes and nodes. The map temperature of all the nodes for a given instant $t = 0\ s$ is known on the entire domain. The discretized heat equation shall be applied over all the nodes of the domain to find the discretization coefficients. In this way, the calculated coefficients are the input parameters of the solver, needed to calculate the temperature map of the next

time instant $t = 0 + \Delta t$. The time instant is then updated and the process is repeated. The detailed solution of the problem can be summarized in the following algorithm:

1. Input data needed:
   - Physical data
   - Geometrical data
   - Boundary Conditions definition
2. Discretization of the domain
3. The initial temperature map is known
4. Update the time instant $t' = t + \Delta t$ and propose a guessed solution for the solver, such as the temperature field of the previous iteration
5. Calculation of the improved temperature field with the preferred solver
6. If $t = 10000\ s$, print the results, otherwise repeat from 4.

3.8.3. Explicit and Implicit methods details

The explicit scheme leads to almost the same equation as Equation 3.6.2, obviously with no heat generation involved:

$$T_P^{n+1} = a_W T_W^n + a_E T_E^n + a_P T_P^n + a_N T_N^n + a_S T_S^n \tag{3.8.2}$$

with the same exact coefficients definition. The boundary conditions are imposed with the methods illustrated in Section 3.7. The only "new" type is the one considered on the right boundary when a sort of Dirichlet boundary condition varying with time is imposed. This condition can be easily handled by applying Equation 3.7.1 at each time step.

The explicit method has its own advantages and disadvantages as explained already before. Briefly, the explicit method is easy to solve because it doesn't require a simultaneous resolution of a set of equations and it is really efficient considering the computational time. Although, it presents restrictive limitation due to the numerical stability, that, in a 2D problem, are even amplified respect to the one-dimensional situation. Indeed, referring to the stability condition 3.6.3, it is clear that reducing the Δx and Δy simultaneously will lead to a even higher reduction of the time step, in order to maintain numerical stability. So that in problems that doesn't require an accurate study of the transient process, it could be rather useless and increase the computational time by orders of magnitude.

The implicit scheme leads, on the other hand, to the same Equation 3.6.4 but, as for the explicit scheme, without the heat generation term:

$$a_S T_S^{n+1} + a_W T_W^{n+1} + a_P T_P^{n+1} + a_N T_N^{n+1} + a_E T_E^{n+1} = T_P^n \tag{3.8.3}$$

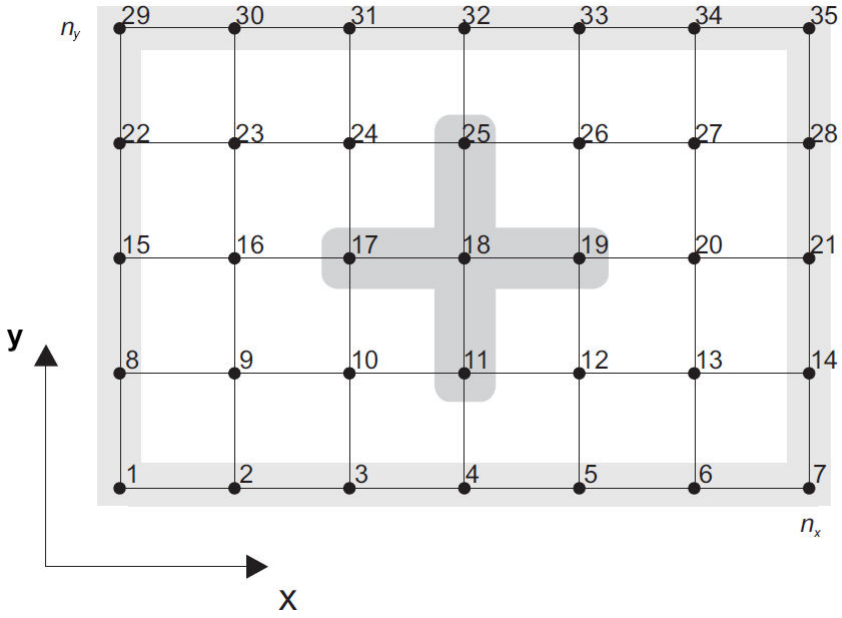


Figure 9: Numbering scheme for a 2D grid [5]

Although the coefficients are the same calculated for the explicit resolution of the problem, the implicit resolution is more complicated since a solver has to be implemented and "book-keeping" issues arise [5]. Two solvers have been implemented: one iterative method, the Gauss-Siedel solver, already discussed, and the Gauss Elimination method achieved through the LU factorization. The LU factorization, or decomposition, factors a matrix as the product of a lower triangular matrix and an upper triangular matrix as already explained in details in Section 2.5.3. It has to be taken into account that the algorithm for the factorization requires $\frac{2}{3}n^3$ floating-point operations, leading to high computational time when a great number of grid points is used. The book-keeping issues refers to the mapping of the $T_{i,j}$ to the entries of a temperature vector $T(k)$, as opposed to the more intuitive matrix $T(i,j)$ that can be used in the explicit scheme. Figure 9 represents one of the possible numbering scheme for a 2D grid.

3.8.4. Assumptions and Expectations

The assumptions considered for this problem can be summarized as:

- Two-dimensional heat diffusion, x and y axis while the z-axis is neglected.
- The thermophysical properties are constant for each material.
- Cell-centered discretization: average values over each cell (temperature, density, specific heat and thermal conductivity).

Considering that this is a pure conduction problem, making expectations on the physical results could be quiet simple and straightforward. There is no heat source in the domain

	$\alpha\left(\frac{m^2}{s}\right)$
M_1	7.423e-05
M_2	5.576e-05
M_3	9.603e-05
M_4	8.951e-05

Table 4: Materials' Thermal Diffusivity

so that all the energy transfer is due to the boundary conditions imposed. Temperature is supposed to arise in the north-east part of the rod's section, considered that on the right side the boundary, the temperature continues to increase over time and on the upper side there is a linear heat flows that impacts on the domain. On the other hand, the temperature field should be at lower values in the south-west area of the domain, where a fixed 18 ^oC temperature is imposed on the bottom part. The rod is in contact with a fluid at $T_g = 35\ ^oC$ on the west side, that, in a first moment of the simulation, will increase the temperature field in the area but, eventually, it could prevent an increase of the temperature when it goes over the 35^oC.

The fact that the properties of the materials are not constant imply that the heat is transferred in a different rate through the domain. In a sense, thermal diffusivity is the measure of thermal inertia [6]. In a substance with high thermal diffusivity, heat moves rapidly through it because the substance conducts heat quickly relative to its heat capacity, meaning that the temperature field will be more "smooth" in the areas of Material 3 and 4, that have the highest α and will be more steep in the other two materials.

3.8.5. Grid Independence study

The mesh chosen for the problem is a uniform rectangular mesh, whose nodes are defined as the intersection of two 1D mesh, one on the x-direction and the other on the y-direction. Sometimes it could be useful to use a non-uniform mesh in order to obtain better results and perform a more efficient simulation. In this problem though, this operation is not necessary due to the relative simplicity and the fact that there is not any focus on a particular section of the domain.

The grid independence study analyzes both the different meshes and the different time-steps used for both the explicit and implicit resolution. The finest mesh achievable is a uniform mesh with $\delta x = \delta y = 5mm$ and $\Delta t = 1\ s$, in compliance with the simulation time of the order of magnitude of few hours maximum.

Although the grid independence study can be performed also for the explicit resolution, it is certainly constrained by the limits of the numerical instability, so that not all the combination of time-steps and mesh refinements make sense. The chosen parameter is the temperature of a point in the north-east region of the domain, with coordinates $x = 0.73\ m$ and $y = 0.62\ m$. Normally, the maximum or minimum temperature is chosen to assess the grid independence but due to the nature of the boundary conditions imposed, these values are fixed

	Implicit - LU				**Implicit - GS**			
	$T_{loc,1}$	$T_{loc,2}$	ε	time [s]	$T_{loc,1}$	$T_{loc,2}$	ε	time [s]
$\delta x = 0.3\ m$	71.00	18.00	3.17e-02	5.76	71.00	18.00	3.09e-02	4.81
$\delta x = 0.1\ m$	60.47	27.78	1.17e-02	14.23	61.53	27.41	1.41e-02	11.21
$\delta x = 0.05\ m$	57.03	31.15	1.01e-03	154.78	57.75	31.14	1.30e-03	124.64
$\delta x = 0.01\ m$	56.13	31.81	3.44e-04	1.40e04	56.01	31.76	3.57e-04	1.13e04

Table 5: Mesh Grid Independence, for the GS solver and LU solver

	Explicit			
	$T_{loc,1}$	$T_{loc,2}$	ε	time [s]
$\delta x = 0.3\ m$	71.00	18.00	2.60e-01	6.52
$\delta x = 0.1\ m$	61.78	27.54	9.31e-02	11.21
$\delta x = 0.05\ m$	58.15	31.33	2.69e-02	25.37
$\delta x = 0.01\ m$	ns	ns	ns	ns

Table 6: Mesh Grid Independence, explicit method

as, respectively, 71^oC and 18^oC. It is preferable to always use a punctual parameter because it could be more subjected to the grid changes and to the solver oscillation, despite some sort of average value that it is not capable to catch this behaviours. In Table 5 the mesh grid independence results are shown, imposing a $\Delta t = 1\ s$ and calculating the numerical relative error as follows:

$$\varepsilon = \frac{||T - T_{ref}||}{||T_{ref}||} \tag{3.8.4}$$

Considering T_{ref} as the best achievable numerical solution, where the the symbolical "$||v||$" represents the Euclidean norm, defined as:

$$||v|| = \sqrt{\sum_{k=1}^{N} |v_k|^2} \tag{3.8.5}$$

where v is a generic vector. It can be noted that for the combination of $\delta x = 0.01\ m$ and $\Delta t = 1\ s$ the explicit method is not stable (ns) due to Equation 3.6.3, so that the simulation should implement a CFL condition, to be checked at each time-step, in order to prevent it to diverge.

The same study is performed on the time-step of the simulation, considering a fixed mesh of $\delta x = \delta y = 5mm$. Results are presented in Table 7.

The grid independence study on the mesh size and the time step refinishing has been performed for both the implicit schemes and the explicit scheme in order to highlight their differences. It can be easily noted that the relative error is higher on the explicit scheme, indicating a slower convergence if compared to the implicit schemes; moreover, with the time-step set at 1 s, the explicit scheme is not stable with a mesh size of 0.01 m or finer. Despite the greater error that is developed using the explicit scheme, it has its point of strength

	Implicit - LU				**Implicit - GS**			
	$T_{loc,1}$	$T_{loc,2}$	ε	time [s]	$T_{loc,1}$	$T_{loc,2}$	ε	time [s]
$\Delta t = 1000s$	47.4	30.02	1.68e-02	24.76	47.10	29.45	1.71e-02	10.56
$\Delta t = 500s$	48.45	30.12	1.57e-02	57.87	47.93	29.59	1.52e-02	21.47
$\Delta t = 100s$	52.74	31.89	4.56e-03	303.12	51.32	30.27	9.32e-03	112.49
$\Delta t = 10s$	56.09	31.75	9.45e-04	3091.6	55.13	31.42	1.89e-03	1147.31

Table 7: Time-Step Grid Independence, comparing LU and GS solver

in the time of execution, especially with the most refined mesh. On the other hand, the time-step analysis could only be performed on the implicit resolutions due to instability issues. The choice of time-step is found to be most relevant, in term of numerical errors; indeed, a relatively big time-step is likely to led to greater errors when compared to the use of less refined mesh. The numerical errors are also shown in Figure 10 and 11 where it is clear that, both for the mesh and the time-step, the error decreases as the number of control volumes or the number of time intervals increase, indicating a successful grid independence of the problem. It has to be taken into account that the "relative numerical error ε" can not be taken as a measure of the overall accuracy of the implicit schemes. Indeed, the comparison is between two completely different solvers: an iterative one (GS) and a direct one (LU). While GS reaches an approximate solution, LU computes the exact solution (in machine precision). It is then useful to compare the solution time with different solver's relative tolerance, calculated as:

$$\varepsilon_s = max\left(\frac{T_i^* - T_i}{T_i}\right) \tag{3.8.6}$$

that is the maximum relative error of a iteration of the solver. In Table 8 the computational time comparison of GS and LU is presented, with different ε_s values. It can be noted that GS computational time is the lowest until $\varepsilon_s = 10^{-10}$ but when an higher accuracy is pursued (such as $\varepsilon_s = 10^{-12}$ and $\varepsilon_s = 10^{-14}$), the computational time of GS increase significantly and the direct solver would be preferable.

	Explicit	**Implicit - LU**	**Implicit - GS**	
Δt $[s]$	time [s]	time [s]	ε_s	time [s]
1	2.74	14.23	10^{-6}	11.21
			10^{-8}	12.39
			10^{-10}	13.46
			10^{-12}	14.26
			10^{-14}	15.37
10	0.05	1.84	10^{-6}	1.36
			10^{-8}	1.52
			10^{-10}	1.71
			10^{-12}	1.89
			10^{-14}	2.06
100	ns	0.55	10^{-6}	0.32
			10^{-8}	0.40
			10^{-10}	0.45
			10^{-12}	0.53
			10^{-14}	0.58

Table 8: Computational time and ε_s of the GS and LU solvers compared to the explicit scheme, with $\delta x = 0.1$ m and several Δt

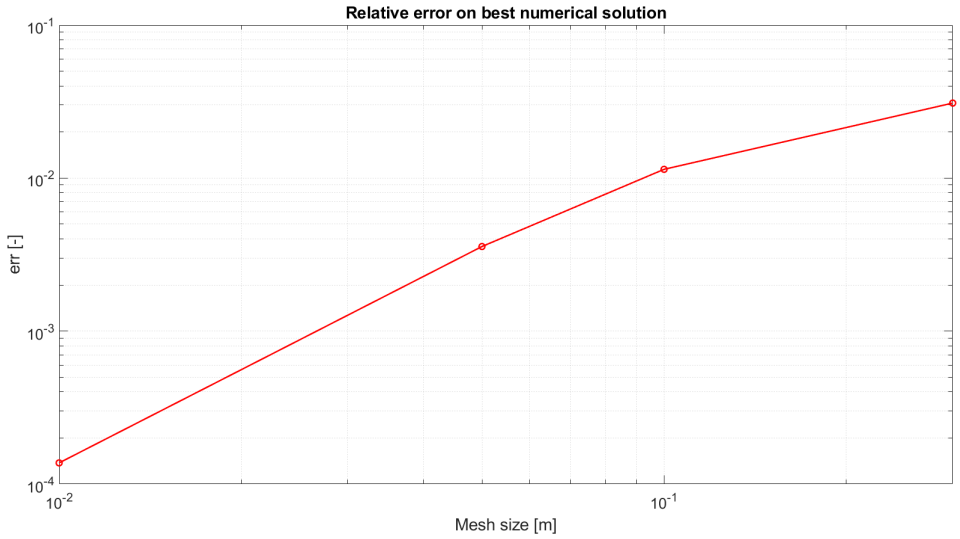


Figure 10: Mesh size grid independence plot

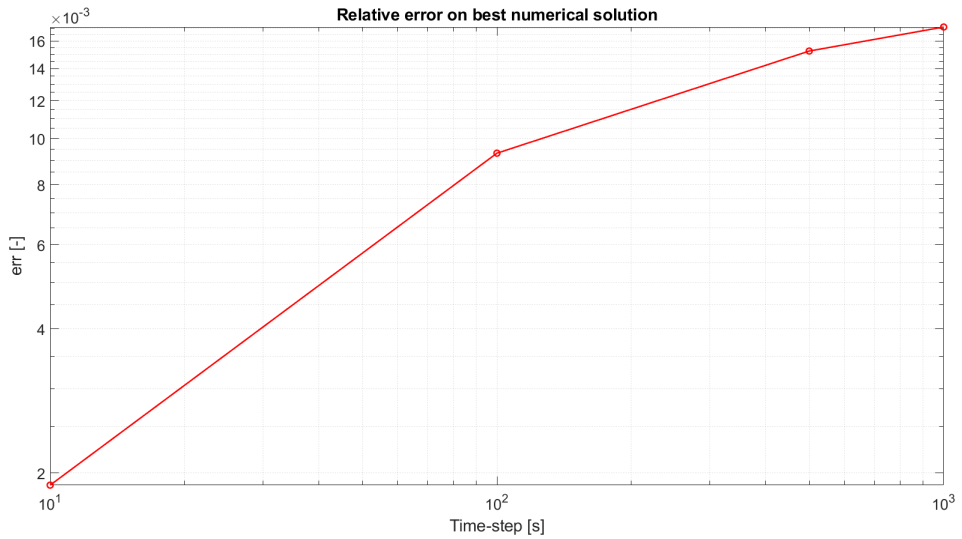


Figure 11: Time step grid independence plot

Therefore, it is quite understandable that for problems that do not require a really accurate resolution (up to machine precision), an iterative solver is preferable due to its faster computation time and satisfying accuracy, but if high accuracy is needed, the iterative solver is not the optimal solution anymore.

3.8.6. Simulation results

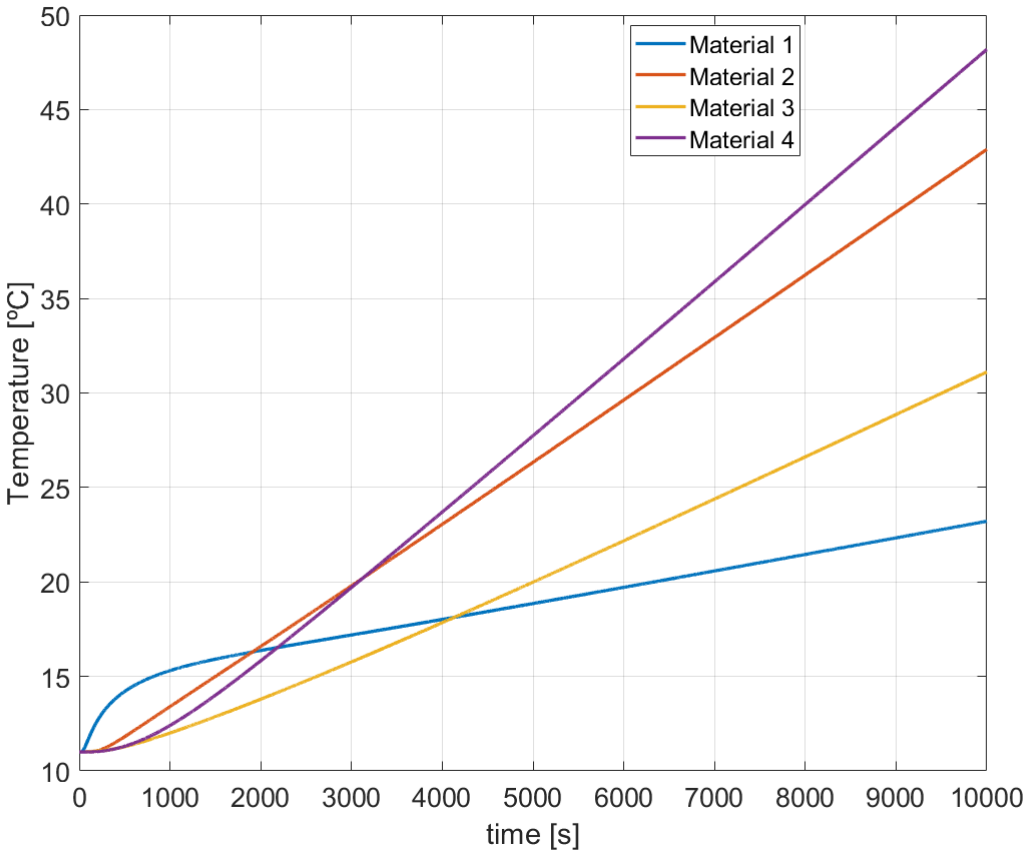
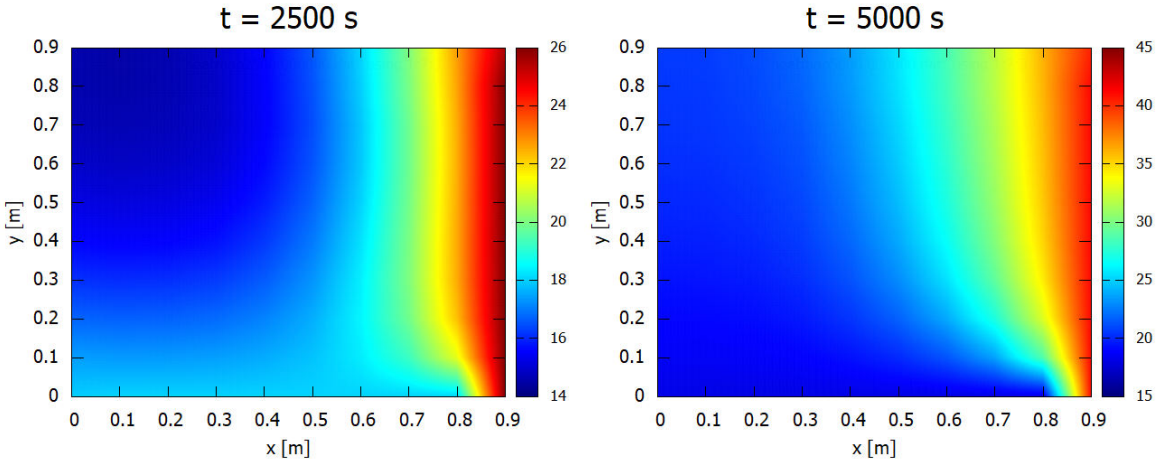


Figure 12: Evolution of temperature of representative nodes

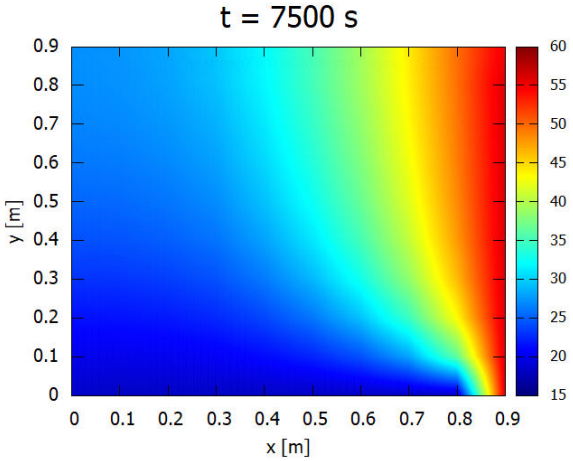
Figure 12 shows the temperature evolution of a representative node for each material of the rod, from $t = 0\ s$ to $t = 10000\ s$. The representative points have been taken as the center of each material although it is not correct to think they represent the uniform evolution of the totality of the material. The evolution of the temperatures can be thought as a superposition of a transient regime from until reaching the steady-state given by left-side, bottom-side and top-side boundary conditions, in addition to a constant perturbation due to the right-side boundary condition. Material 3 and Material 4 have the lower thermal inertia (defined as $I = \sqrt{k\rho c_p}$); this explains why they would last less to reach the fictional steady-state, while Material 1 and Material 2 (especially Material 1) present a clear transient regime before reaching the linearity.

The simulation overall results are presented as timestamps of the transient process at the times t = 2500 s, 5000 s, 7500 s and 10000 s as shown in Figure 13.

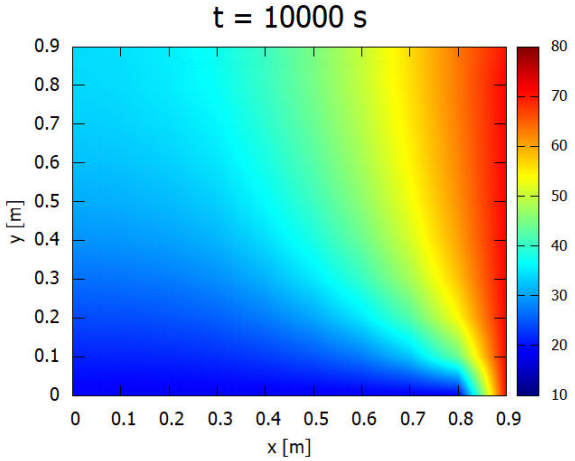


(a) Timestamp at t = 2500 s

(b) Timestamp at t = 5000 s



(c) Timestamp at t = 7500 s



(d) Timestamp at t = 10000 s

Figure 13: Simulation results at different timesteps

The overall results are as expected, showing the highest temperatures in the right upper part and lowest temperatures in the left lower region. The heat transmission rate is higher in the upper region due to the heat flow impacting on the top part of the domain and this cause the isotherms curves to be closer in the right-lower angle and gradually thinned out towards the opposite angle. The timestamps have been taken from the implicit resolution through the use of the Gauss-Siedel iterative method. Despite the LU factorization solver is equally valid for the resolution of this problem, the iterative solar is preferable for its higher computational speed, particularly when the mesh becomes more refined.

4. Convection and diffusion

This section shows how the FVM is applied to a model of convective transport: the 2D convection-diffusion equation. First of all this section introduces and compares the numerical schemes adopted for the convective term in transport equations, then some benchmark problems are presented, solved and discussed. The convection-diffusion equation is a model of transport of heat, mass and other passive scalars (Φ). Convection is created by fluid flow and the goal of this section is to obtain a solution for Φ in the presence of a given fluid flow. The fluid flow knowledge is not discussed at this point, it could have come from an experiment, be given as analytical solution or simply be guessed. The origin of the flow field is irrelevant in this discussion[2]. Applying the FVM to this equation allows different schemes for approximating the convection term to be compared. This chapter should be considered a brief introduction to the topic of convection modeling schemes. The schemes considered are the upwind difference scheme (UDS), the central difference scheme (CDS), the exponential difference scheme (EDS), the hybrid scheme and the power-law scheme.

4.1. The convection-diffusion equation

The general PDE is:

$$\frac{\partial(\rho\Phi)}{\partial t} + \nabla\cdot(\rho\vec{v}\Phi) = \nabla\cdot(\Gamma\nabla\Phi) + S \tag{4.1.1}$$

where $\vec{v}$ is the given flow field, Γ is the diffusion coefficient and S the source term. The quantities Γ and S refers to a particular meaning of the general variable Φ. The equation can be seen as a sum of four different terms (from left to right): the unsteady term, the convection term, the diffusion term and the source term. Moreover, it should be remembered that the word "diffusion" is used in a generalized sense. The diffusion flux due to the gradient of the general variable Φ is $-\Gamma\frac{\partial\Phi}{\partial x_j}$ [2] which, for different meaning of Φ, could represent chemical-species diffusion flux, heat flux, viscous stress, etc. Actually the expressions $\frac{\partial}{\partial x_j}$ denotes the sum of three terms for the three coordinate directions. In the end, since the given flow field has to satisfy the continuity equation:

$$\frac{\partial\rho}{\partial t} + \nabla\cdot(\rho\vec{v}) = 0 \tag{4.1.2}$$

the general PDE can also be written as:

$$\rho\frac{\partial\Phi}{\partial t} + \rho u_j\frac{\partial\Phi}{\partial x_j} = \frac{\partial}{\partial x_j}\left(\Gamma\frac{\partial\Phi}{\partial x_j}\right) + S \tag{4.1.3}$$

From this form of the equation, it follows that, for given distribution of ρ, u_j, Γ and S, any solution Φ would both satisfy both the equations.[2]

4.2. Steady state one-dimensional convection and diffusion

As it hass been done for the chapter about heat diffusion, much can be learned from the consideration of the simplest case. So that a steady one-dimensional situation with no sources is considered; the governing differential equation is:

$$\frac{\mathrm{d}}{\mathrm{d}x}(\rho u\Phi) = \frac{\mathrm{d}}{\mathrm{d}x}\left(\Gamma\frac{\mathrm{d}\Phi}{\mathrm{d}x}\right) \tag{4.2.1}$$

where u is the velocity in the x direction. Also, the continuity equation becomes:

$$\rho u = constant \tag{4.2.2}$$

In order to derive the discretization equation, the grid cluster shown in Figure 14 is similar

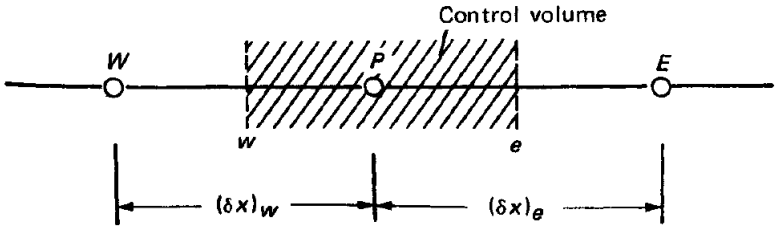


Figure 14: One-dimensional grid point cluster [2]

to the one introduces in the last section. It is convenient to assume that e is located midway between P and E. The different discretization schemes are discussed in the next paragraphs.

4.2.1. CDS

Integration of Equation 4.2.1 over the control volume of Figure 14 leads to:

$$(\rho u\Phi)_e - (\rho u\Phi)_w = \left(\Gamma\frac{\mathrm{d}\Phi}{\mathrm{d}x}\right)_e - \left(\Gamma\frac{\mathrm{d}\Phi}{\mathrm{d}x}\right)_w \tag{4.2.3}$$

The diffusion term is discretized as already shown in the heat diffusion chapter and, for the convection term, the same choice of profile would at first seem natural, resulting:

$$\Phi_e = \frac{1}{2}(\Phi_E + \Phi_P) \quad and \quad \Phi_w = \frac{1}{2}(\Phi_P + \Phi_W) \tag{4.2.4}$$

The factor $\frac{1}{2}$ arises from the assumption of the interfaces being midway. Now, Equation 4.2.3 can be written as:

$$\frac{1}{2}(\rho u)_e(\Phi_E + \Phi_P) - \frac{1}{2}(\rho u)_w(\Phi_P + \Phi_W) = \frac{\Gamma_e(\Phi_E - \Phi_P)}{(\delta x)_e} - \frac{\Gamma_w(\Phi_P - \Phi_W)}{(\delta x)_w} \tag{4.2.5}$$

The values Γ_e and Γ_w have to be obtained in the same way in the previous chapter k_e and k_w were obtained. In the end, to arrange the equation more compactly, the new two symbols F and D are introduced as follows:

$$F \equiv \rho u, \quad D \equiv \frac{\Gamma}{\delta x} \tag{4.2.6}$$

Both have the same dimensions; F indicates the strength of the convection (or flow), while D is the diffusion conductance. It should be noted that, whereas D remains always positive, F can take positive or negative values depending on the direction of the flow. With the new symbols introduced:

$$a_P \Phi_P = a_E \Phi_E + a_W \Phi_W \tag{4.2.7}$$

where

$$a_E = D_e - \frac{F_e}{2}$$

$$a_W = D_w + \frac{F_w}{2}$$

$$a_P = D_e + \frac{F_e}{2} + D_w - \frac{F_w}{2} = a_E + a_W + (F_e - F_w)$$

4.2.2. UPS

The upwind scheme recognizes that a weak point in the CDS scheme is the assumption that the convected property Φ_e at the interface is the average of Φ_E and Φ_P, and it proposes a different description. In the UPS scheme, the value of Φ at an interface is equal to the value of Φ at the grid point on the *upwind* side of the face, so that:

$$\Phi_e = \Phi_P \quad if \quad F_e > 0, \tag{4.2.8}$$

$$\Phi_e = \Phi_E \quad if \quad F_e < 0$$

The value of Φ_w can be similarly defined. Replacing then Equation 4.2.4 with this concept, the discretization equation becomes:

$$a_P \Phi_P = a_E \Phi_E + a_W \Phi_W \tag{4.2.9}$$

where

$$a_E = D_e + max(-F_e, 0)$$

$$a_W = D_w + max(F_w, 0)$$

$$a_P = D_e + max(F_e, 0) + D_w + max(-F_w, 0) = a_E + a_W + (F_e - F_w)$$

where the operator $max(A,B)$ denotes the greater of A and B. The essence of the UPS can be explained referring to the "tank-and-tube" model (Gosman,Pun et al, 1969). As shown

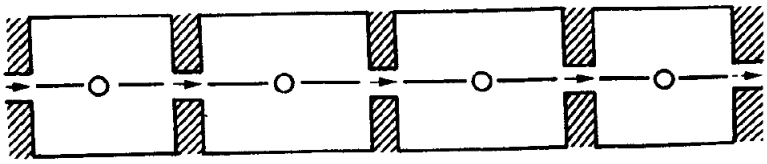


Figure 15: Tank-and-tube model

in Figure 15, the control volumes can be thought to be stirred tanks that are connected by short and thin tubes. The flow through the tubes represents convection while the conduction through the tank represents diffusion; each tank contains a uniform temperature fluid. Therefore, it is appropriate to assume that the fluid flowing in each connecting tube has the temperature that prevails in the *upstream* side; the fluid is not able to know anything about the tank toward which it is heading, but would carry the full legacy of the tank from which it has come [2].

4.2.3. The exact solution

The governing Equation 4.2.1 can fortunately be solved exactly when Γ is taken as a constant. If a domain $0 \leq x \leq L$ is used, with the boundary conditions:

$$At \quad x = 0 \quad \Phi = \Phi_0, \tag{4.2.10}$$

$$At \quad x = L \quad \Phi = \Phi_L$$

the solution of Equation 4.2.1 is:

$$\frac{\Phi - \Phi_0}{\Phi_L - \Phi_0} = \frac{exp(Px/L) - 1}{exp(P) - 1} \tag{4.2.11}$$

where P is the Peclet number, defined by:

$$P \equiv \frac{\rho u L}{\Gamma} \tag{4.2.12}$$

The Peclet number can be seen as the ratio of the strength of convection and diffusion. The behaviour of the exact solution can be understood from Figure 16, where the variation $\Phi \sim x$ has been plotted for different values of the Peclet number. In the limit of zero Peclet number, the $\Phi \sim x$ variation is linear because the problem reduces to a pure diffusion (or conduction) problem. When the flow is in the positive x direction ($P > 0$), the values of Φ in the domain seem to be more influenced by the upstream value Φ_0. As the Peclet number increases, the values of Φ tend to remain very close to the upstream value. The situation is reversed if the flow is towards the negative x direction. Now, the assumption made when deriving the UPS scheme seem to be correct although it is used for all the values of P, not just for large values.

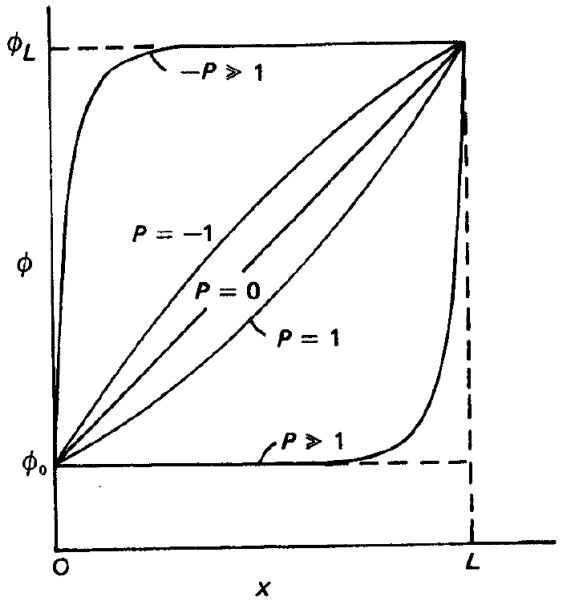


Figure 16: Exact solution for the one-dimensional convection-diffusion problem [2]

Moreover, the UPS always calculates the diffusion term from a linear $\Phi \sim x$ profile and thus overstimates diffusion at large values of P.

4.2.4. Outline of other methods

In this section, other 3 methods are discussed but the mathematical details are not presented. The details can be found in [2].

The exponential scheme

A total flux J is defined as the sum of the convection flux $\rho u\Phi$ and the diffusion flux $-\Gamma d\Phi/dx$. Thus,

$$J = \rho u\Phi - \Gamma\frac{\mathrm{d}\Phi}{\mathrm{d}x} \tag{4.2.13}$$

With this definition, Equation 4.2.1 becomes:

$$\frac{\mathrm{d}J}{\mathrm{d}x} = 0 \tag{4.2.14}$$

This equation can be integrated over the control volume and the exact solution 4.2.11 can be used as a profile between points P and E. With further mathematical manipulation a discretized equation can be cast into the standard form:

$$a_P\Phi_P = a_E\Phi_E + a_W\Phi_W \tag{4.2.15}$$

where

$$a_E = \frac{F_e}{exp(F_e/D_e) - 1}$$

$$a_W = \frac{F_w exp(F_w/D_w)}{exp(F_w/D_w) - 1}$$

$$a_P = a_E + a_W + (F_e - F_w)$$

When used for the one-dimensional problem, this scheme gives the exact solution for any Peclet number of grid size.

The Hybrid Scheme

The hybrid scheme was developed by Spalding (1972). The hybrid scheme is constructed in connection with the exponential scheme. In order to appreciate this connection, the dimensionless form a_E/D_e is plotted in Figure 17, as a function of the Peclet number P_e Certain

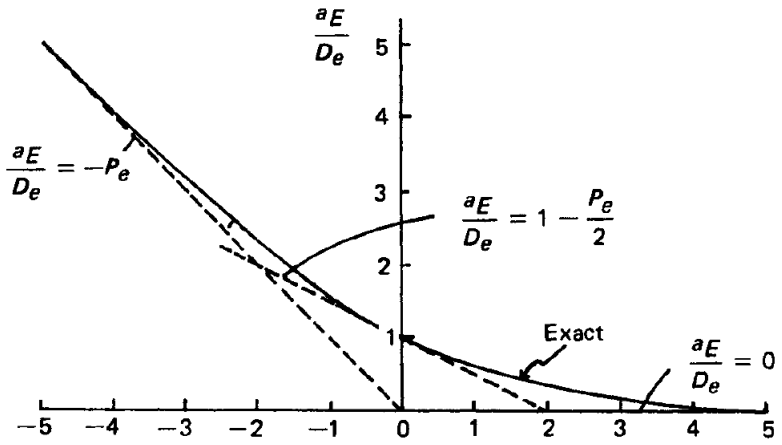


Figure 17: Variation of the coefficient a_E/D_e with Peclet number [2]

specific properties of the exact variation or a_E/D_e can be noted:

For $P_e \to +\infty$,

$$\frac{a_E}{D_e} \to 0,$$

For $P_e \to -\infty$,

$$\frac{a_E}{D_e} \to -P_e,$$

At $P_e = 0$,

$$\frac{a_E}{D_e} = 1 - \frac{P_e}{2}$$

The three straight lines represented on the figure are this limiting cases and they can be seen as a reasonable approximation to the exact curve so that a discretized equation is derived and written in the following compact way:

$$a_P \Phi_P = a_E \Phi_E + a_W \Phi_W \tag{4.2.16}$$

where

$$a_E = max(-F_e, D_e, -\frac{F_e}{2}, 0)$$

$$a_W = max(-F_w, D_w, \frac{F_w}{2}, 0)$$
$$a_P = a_E + a_W + (F_e - F_w)$$

The Power-Law Scheme

It can be seen from Figure 17 that the hybrid scheme really differs from the exact solution at $P_e = \pm 2$ and it seems quite premature to set the diffusion equal to zero as soon as $|P_e|$ exceeds 2 [2]. The scheme, developed by Patankar (1979), provide a better representation of the exponential behaviour. The parameter a_E/D_e can be written in a compact way as:

$$a_E = D_e \cdot max\left(0, \left(1 - \frac{0.1|F_e|^5}{D_e}\right)\right) + max(0, -F_e) \tag{4.2.17}$$

4.2.5. A Generalized Formulation

A generalized formulation is developed in Patankar (1979) where all the schemes presented can now be thought of as merely different choice of a function $A(|P|)$:

$$a_P\Phi_P = a_E\Phi_E + a_W\Phi_W \tag{4.2.18}$$

where

$$a_E = D_e A(|P_e|) + max(-F_e, 0)$$
$$a_W = D_w A(|P_w|) + max(F_w, 0)$$
$$a_P = a_E + a_W + (F_e - F_w)$$

Expressions for $A(|P|)$ are listed in Table 9 Before leaving the one-dimensional situation, a

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

Table 9: Function $A(|P|)$ for different schemes

brief comparison between the schemes is presented. The value of a grid point Φ_P is predicted by the various schemes for given values of $\Phi_E = 1$ and $\Phi_W = 0$ and plotted for various values of P in Figure 18 All the schemes, excepted the CDS, give a physically realistic solution inside the range 0-1 established by the imposed values. Since it is the grid Peclet number that influences the behaviour of these numerical models, the grid could, in principle, be refined until P is small enough (<2) for the CDS scheme to yield reasonable solutions.

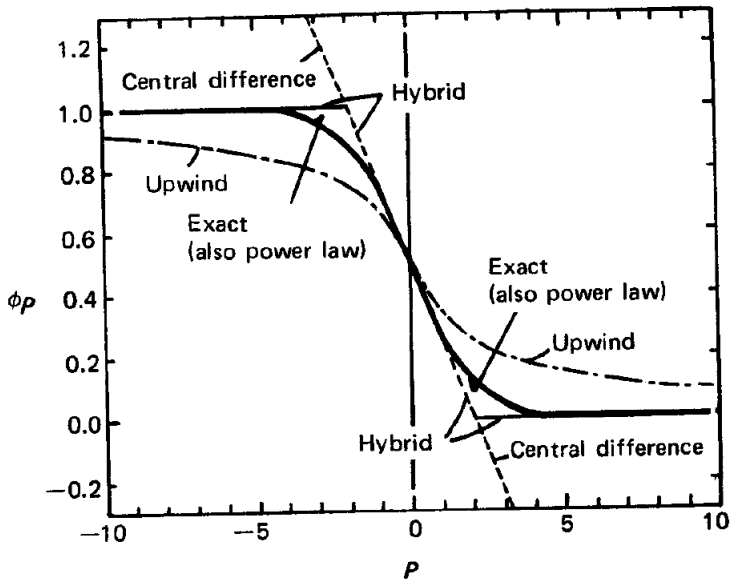


Figure 18: Prediction of Φ_P by the various schemes for a range of Peclet numbers [2]

However, an excessively fine grid could not be feasible and, in any case, it is really better to pursue a numerical model that gives physically realistic solution even for coarse grids.

4.3. Discretization Equation for two-dimensional problems

All the one-dimensional discussion can be properly extended to a two-dimensional situation, referring to the control volume of Figure 19. The details of the derivation are not given in this work but they can be found again in the literature [2].

The two-dimensional discretization equation can be written as:

$$a_P\Phi_P = a_E\Phi_E + a_W\Phi_W + a_N\Phi_N + a_S\Phi_S + b \tag{4.3.1}$$

where:

$$a_E = D_eA(|P_e|) + max(-F_e, 0),$$

$$a_W = D_wA(|P_w|) + max(F_w, 0),$$

$$a_N = D_nA(|P_n|) + max(-F_n, 0),$$

$$a_S = D_sA(|P_s|) + max(F_s, 0),$$

$$a_P^0 = \frac{\rho_P^0\Delta x\Delta y}{\Delta t},$$

$$b = S_C\Delta x\Delta y + a_P^0 + \Phi_P^0,$$

$$a_P = a_E + a_W + a_N + a_S + a_P^0 - S_P\Delta x\Delta y$$

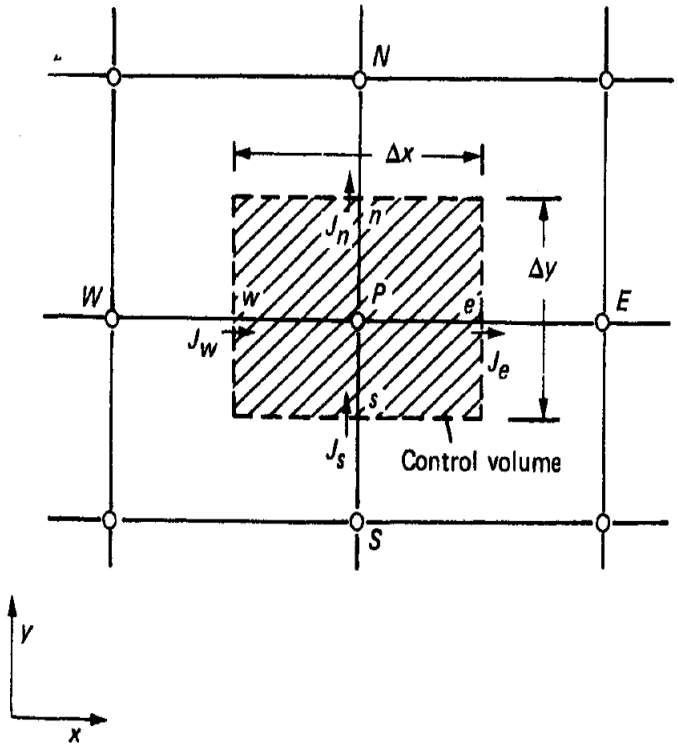


Figure 19: Control Volume for the two-dimensional situation [2]

The function $A(|P|)$ can be selected from Table 9 for the desired scheme. Some physical understanding of the discretized equation can be given even at this stage. The neighbor coefficients a_E, a_W, a_N, and a_S represent the convection and diffusion influence at the four surfaces of the CV, in terms of the flow rate F and the conductance D. The term $a_P^0\Phi_P^0$ can be thought of as the known Φ content inside the control volume at the time t, divided by the time step.

The three-dimensional discretization will not be presented because it is out of scope for this work.

4.4. Benchmark Problems

Two simple but explanatory problems are presented in this section. The difference schemes differences will arise and the false diffusion topic will be introduced.

4.4.1. Bidimensional flow with a unidimensional variation of the variable Φ solved in the same direction of the flow

. This problem is the simplest possible but it can be useful to validate the convection-diffusion equation, given a fluid field. Referring to Figure 20, the given fluid field is:

$$u(x,y) = U_0$$

$$v(x,y) = 0$$

The governing differential equation is 4.2.1 with no sources and the steady state analyt-

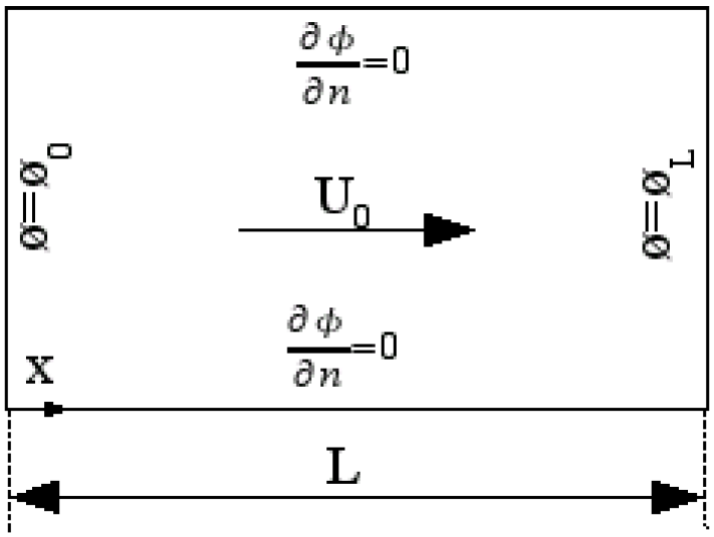
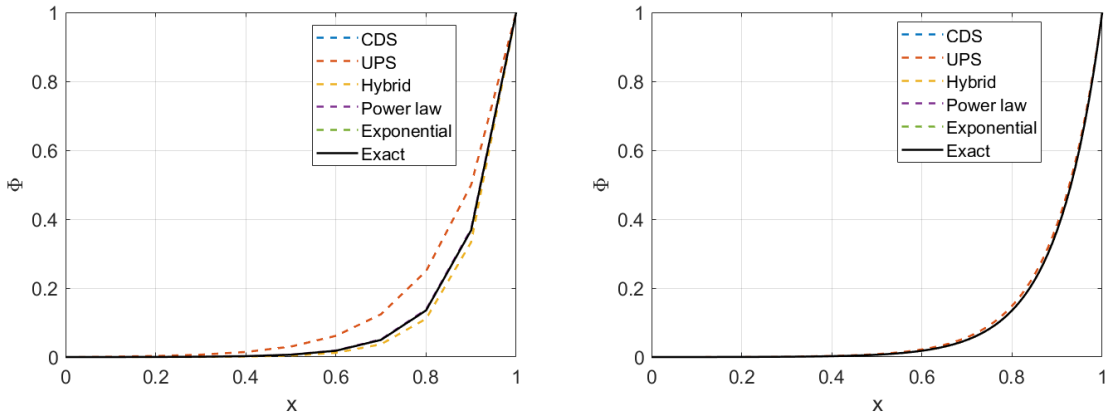


Figure 20: Schematics of the problem

ical solution is the same as Equation 4.2.11. The problem is solved with the numerical techniques presented in the past sections and all the methods described are compared in Figure 21. In order to have a better understanding of the methods, a coarse mesh with



(a) $\Delta x = 0.1$ *(b)* $\Delta x = 0.01$

Figure 21: Solution of the problem with different numerical schemes

$\Delta x = (\delta x)_e = (\delta x)_w = 0.1$ and a finer mesh $\Delta x = (\delta x)_e = (\delta x)_w = 0.01$ are implemented. The coarse mesh highlights properly the differences of the schemes: the CDS, power-law and exponential schemes are indistinguishable from the exact solution (indeed, for a one-dimensional problem, the exponential scheme is the exact solution) while the hybrid and the UPS scheme shows a lower convergence. Details of the numerical error can be found in Table 10, calculated with Equation 3.8.4 with the due differences; in this case the reference

value is the exact analytical solution. As expected, UPS scheme has the higher errors then

	ε ($\Delta x = 0.1$)	ε ($\Delta x = 0.01$)
UPS	1.82e-01	3.27e-02
CDS	4.14e-02	5.61e-04
Hybrid	4.14e-02	5.61e-04
Power law	4.17e-03	1.06e-04
Exponential	7.64e-17	6.97e-15

Table 10: Relative error of the numerical schemes

the other schemes. It can also be noted that, when refining the mesh, the numerical errors decrease for every scheme (the exponential scheme error is practically the machine precision). The power law scheme results to be the most accurate scheme so far.

4.4.2. Unidimensional flow with a unidimensional variation of the variable solved in the perpendicular direction of the flow

It will be immediately clear that this situation is the same as a conduction in a moving solid, independently of the velocity field. Indeed, for this problem the velocity field is:

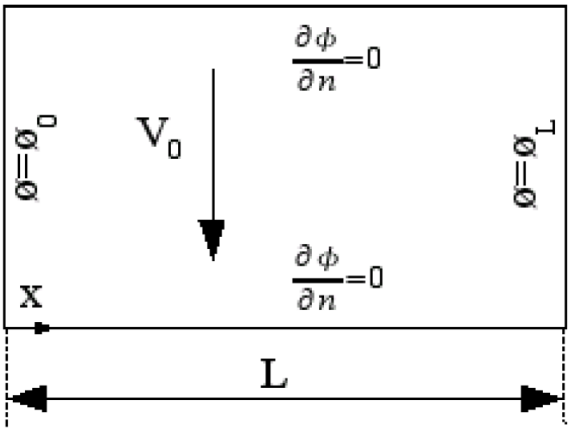


Figure 22: Schematics of the problem

$$u(x,y) = 0$$

$$v(x,y) = V_0$$

and the exact analytical solution is

$$\Phi = \Phi_0 + \frac{\Phi_L - \Phi_0}{L} x \tag{4.4.1}$$

It is straightforward to understand that, since $u = 0$ and being a one-dimensional flow, equation 4.2.1 becomes:

$$\frac{\mathrm{d}}{\mathrm{d}x}\left(\Gamma\frac{\mathrm{d}\Phi}{\mathrm{d}x}\right) = 0 \tag{4.4.2}$$

and the problem becomes a pure diffusion one where Γ has the same role that the thermal conductivity had in the previous sections. Therefore the problem has been solved with the same procedure of the first benchmark problem. Results are provided in Figure 23. As

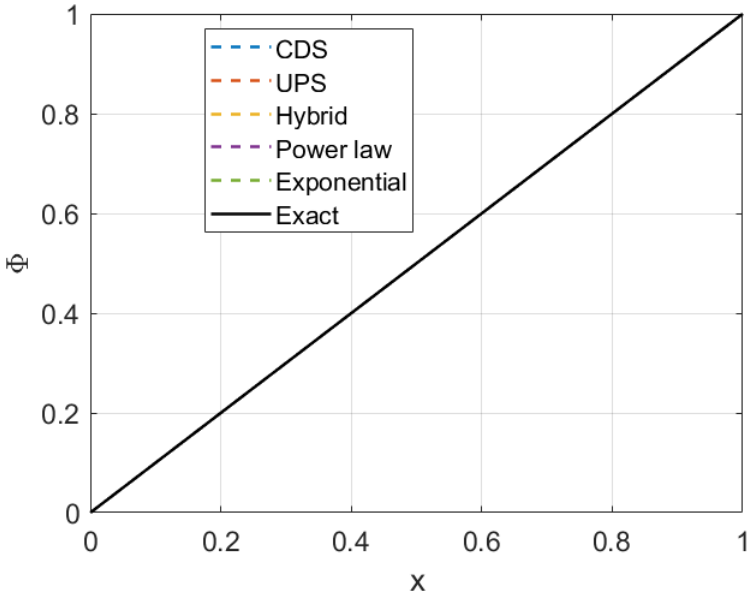


Figure 23: Solution of the problem with different numerical schemes

expected, the solution is linear and all the schemes are able to perfectly predict the analytical solution.

4.4.3. Two-dimensional Diagonal Flow

The first two-dimensional problem is diagonal flow in the main diagonal of a square domain. The velocity field can be represented as:

$$u(x,y) = V_0 \cdot cos(\alpha)$$

$$v(x,y) = V_0 \cdot sin(\alpha)$$

For this kind of flow, the solution is known for an infinite total Peclet number:

$$\Phi = \Phi_1 \quad above\ the\ diagonal \tag{4.4.3}$$

$$\Phi = \Phi_2 \quad below\ the\ diagonal$$

It's the first time that the flow is oblique to the grid lines and that there is a nonzero gradient

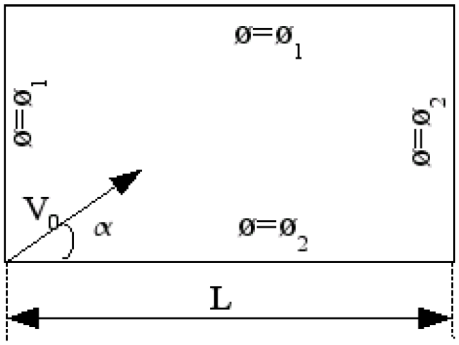


Figure 24: Schematics of the problem

of the dependant variable in the direction of the flow. This problem highlight properly the matter of the "False Diffusion". If the coefficients of Equation 4.2.7 and Equation 4.2.9 are compared, it can be shown that the UPS scheme is the equivalent to replacing Γ in the CDS scheme with $\Gamma + \rho u \delta x/2$. So that the upwind scheme seems to augment the true diffusion coefficient Γ with a fictitious (false) diffusion coefficient $\rho u \delta x/2$. Therefore, it could be easy to conclude that the CDS scheme, being a second-order accuracy scheme, is more accurate than the upwind scheme. In fact, the matter is more complicated. It can be shown [2] that the so-called false diffusion coefficient $\rho u \delta x/2$ could be a desirable at large Peclet numbers, because it actually corrects the wrong implications that occurs from the central-difference scheme. In any case, the matter of false diffusion is never serious at low Peclet number because the real diffusion is large in comparison; on the other hand, at large Pecler numbers, the false diffusion is more important and it should be taken into account.

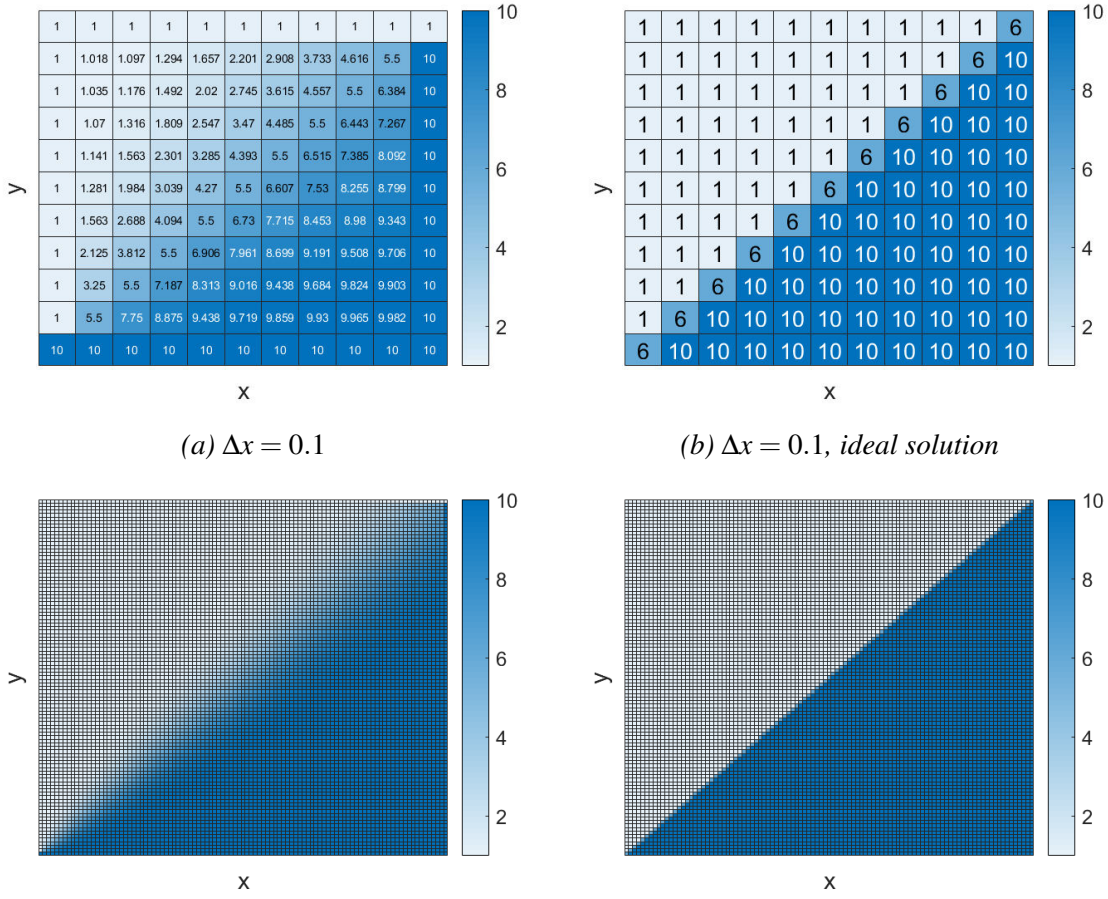
The diagonal flow is hence solved again using a coarse ($\delta x = \delta y = 0.1$) mesh and a refined ($\delta x = \delta y = 0.01$) mesh in order to study the solution sensitivity to the mesh. The problem is solved with a large Peclet number ($Pe \to +\infty$) with a diffusion coefficient that tends to 0. Moreover, two schemes have been used, CDS in results are shown in Figure 25 and UPS in Figure 26 where they are compared with the ideal solution with $\Gamma = 0$.

It is immediately clear that the CDS scheme provides more accurate results for both the coarse and fine mesh. From Table 11 it is clear that CDS has the smaller numerical error while all the other schemes present the same higher error due to the order of convergence of the schemes. The UPS scheme suffers the false diffusion due to the scheme formulation itself

	ε ($\Delta x = 0.1$)	ε ($\Delta x = 0.01$)
CDS	6.90e-02	7.54e-03
UPS	1.34e-01	6.72e-02
Hybrid	1.34e-01	6.72e-02
Power law	1.34e-01	6.72e-02
Exponential	1.34e-01	6.72e-02

Table 11: Relative error of the numerical schemes

and the fact that the flow is oblique to the grid lines, as explained before. An approximate



(a) $\Delta x = 0.1$ *(b)* $\Delta x = 0.1$, *ideal solution*

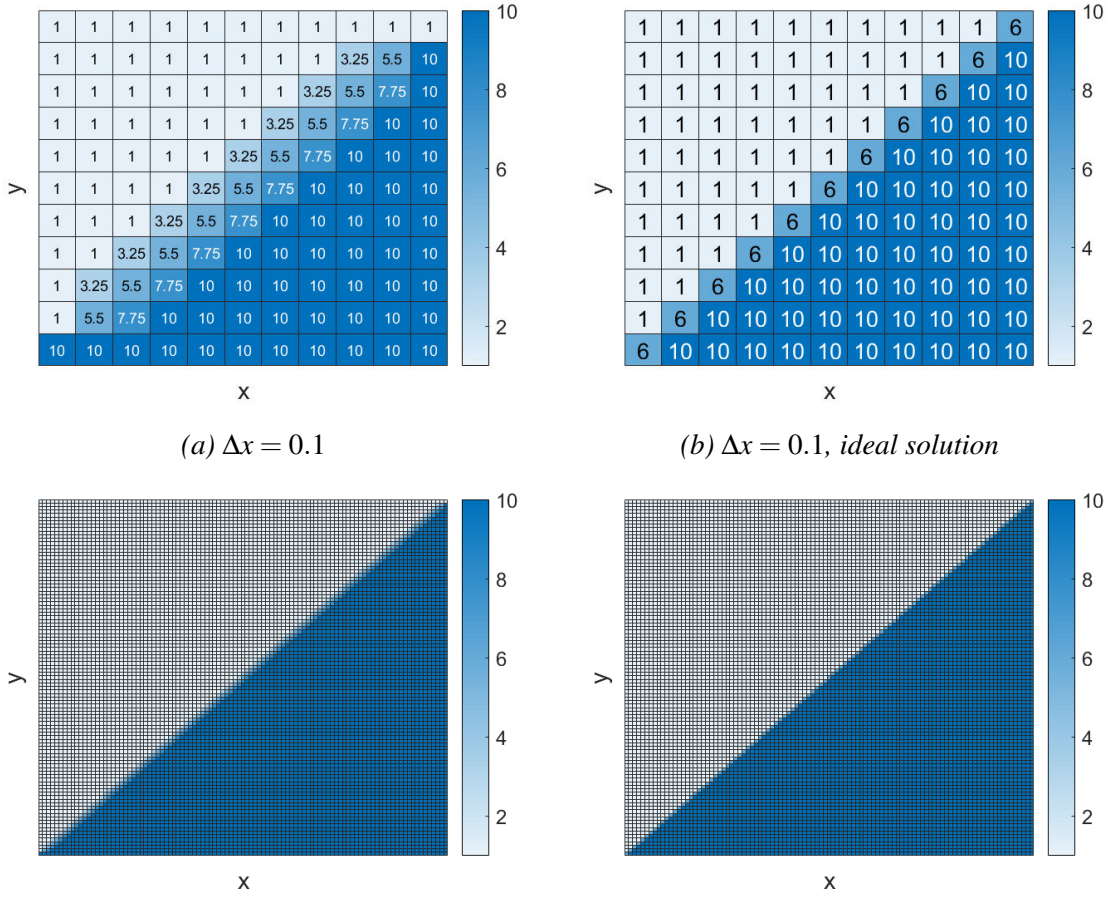
(c) $\Delta x = 0.1$, *ideal solution* *(d)* $\Delta x = 0.01$, *ideal solution*

Figure 25: Solution of the problem with UPS scheme

expression for the false diffusion coefficient for a two-dimensional situation has been given by Vahl Davis and Mallinson (1972):

$$\Gamma_{false} = \frac{\rho U \Delta x \Delta y \, sin2\theta}{4(\Delta y \, sin^3\theta + \Delta x \, cos^3\theta)} \tag{4.4.4}$$

where U is the resultant velocity, and θ is the angle (between 0 and 90°) made by the velocity vector with the x direction. With a real $\Gamma = 0$, Γ_{false} has been found equal to 35.3 for the coarse mesh and 3.53 for the fine mesh. This explains how the solution is really sensitive to the grid. In order to reduce the false diffusion the grid should be as refined as possible and the grid should align with the fluid flow. Schemes that would give less false diffusion have been worked out [10] but their discussion, considering the more complexity, is out of scope for this work.



(a) $\Delta x = 0.1$ *(b)* $\Delta x = 0.1$, *ideal solution*

(c) $\Delta x = 0.1$, *ideal solution* *(d)* $\Delta x = 0.01$, *ideal solution*

Figure 26: Solution of the problem with CDS scheme

4.5. The Smith-Hutton problem

In a well-known paper published in 1982, Smith and Hutton presented results of several authors' attempts to numerically solve a specially devised test problem involving streamline curvature typical of recirculating flows and steep variations in the transported scalar [7]. This problem [Smith Hutton, 1982] is defined, solved and discussed next.

4.5.1. Problem definition

The two-dimensional test problem devised by Smith and Hutton is concerned with steady-state convection and diffusion of a scalar field, defined by the general variable Φ in a prescribed velocity field. The governing equation is:

$$\frac{\partial(\rho\Phi)}{\partial t} + \nabla\cdot(\rho\vec{v}\Phi) = \nabla\cdot(\Gamma\nabla\Phi) \tag{4.5.1}$$

without the internal source. A solenoidal velocity field is given as:

$$u(x,y) = 2y(1-x^2) \tag{4.5.2}$$

$$v(x,y) = -2x(1-y^2)$$

that produces the pattern of streamlines shown in Figure 27. The following boundary condi-

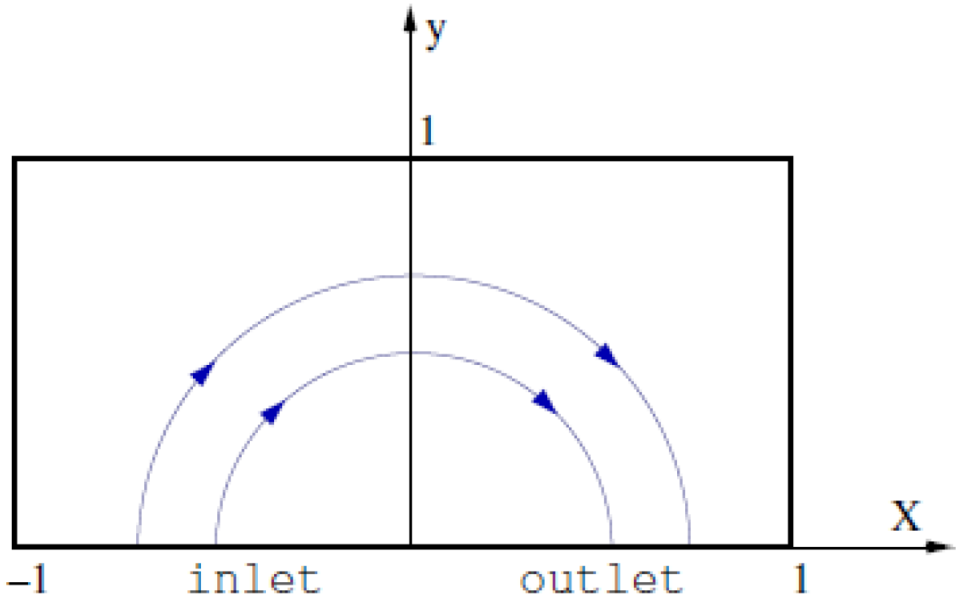


Figure 27: Scheme of the Smith-Hutton problem

tions are imposed for the variable Φ, where $\alpha = 10$:

$$\Phi = 1 + tanh[(2x+1)\alpha] \qquad y = 0;\ x \in (-1,0)\ \ (inlet) \tag{4.5.3}$$

$$\frac{\partial \Phi}{\partial y} = 0 \qquad y = 0;\ x \in (0,1)\ \ (outlet)$$

$$\Phi = 1 - tanh(\alpha) \qquad (elsewhere)$$

Therefore Φ is essentially 0 everywhere on $x = 0$ except close to the origin of the coordinates where $\Phi = 2$. The problem is solved for a range of Peclet numbers (10, 10^3,10^6) and it is verified with another numerical solution given by CTTC center of Terrassa, Spain.

4.5.2. Algorithm development

The following assumptions has been made in the resolution of the problem:

- Steady state
- Pure two-dimensional flow where the z-axis flow is neglected.
- Constant thermophysical properties,
- FVM with cell-centered discretization and average values over each cell.

The domain is discretized in small rectangles and the nodes are placed in the center of each control volume. Thus, the discretized equation is:

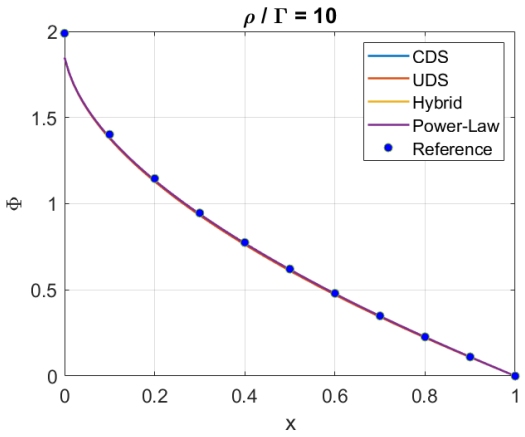
$$a_P\Phi_P = a_E\Phi_E + a_W\Phi_W + a_N\Phi_N + a_S\Phi_S \tag{4.5.4}$$

with the same coefficient of Equation 2.3.1. Obviously, the source term is not present. After obtaining all the discretization coefficients, the property map is calculated employing the GS solver. The proposed algorithm for the resolution of the Smith-Hutton problem is presented below.

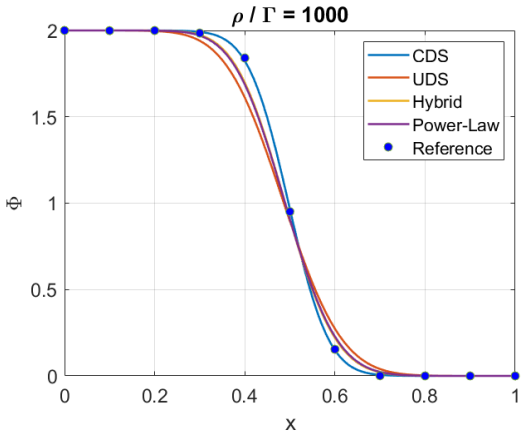
1. Problem input data:
    - Physical data (ρ,Γ)
    - Geometrical data
    - Numerical data and boundary conditions
    - Choice of the convective scheme.
2. Guess of the property field $\Phi[i, j]$ needed by the solver.
3. Calculation of the discretization coefficient according to the chosen scheme.
4. Calculation of the property map $\Phi[i, j]$ using the GS solver
5. Print the results

4.5.3. Verification

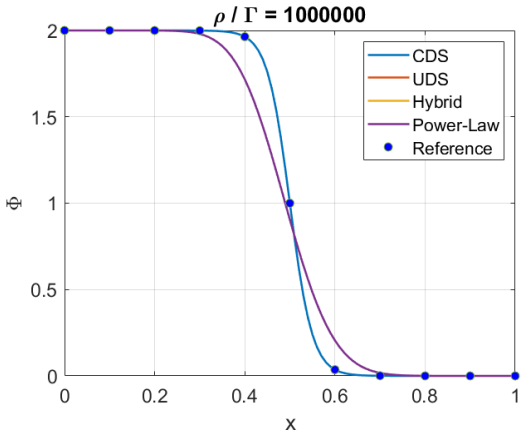
The problem is solved with the usual schemes for convection and the results are validated with a reference solution provided by the CTTC. In particular, the properties at the outlet are compared in the following figures. The problem has been solved with a regular mesh of 200x100 nodes.



(a) Outlet property values with $\rho/\Gamma = 10$



(b) Outlet property values with $\rho/\Gamma = 10^3$



(c) Outlet property values with $\rho/\Gamma = 10^6$

Figure 28: Outlet property values with different Peclet numbers

It can be seen from Figure 28 that, for $\rho/\Gamma = 10$, all the numerical schemes appear to behave in a similar way and that they encounter some difficulty on predict the value at $x = 0$. For $\rho/\Gamma = 10^3$ some differences among the schemes start to arise and it can be noted that the CDS is the best performing one, followed by the Power-law scheme and secondly by UDS and the hybrid scheme. The CDS scheme leads to more accurate results for all the reason explained in section 4.4.3 while the other schemes are affected more by the false diffusion.

The results for $\rho/\Gamma = 10^6$ behave similarly to the previous situation, having better results using the CDS scheme. In Tables 12, 13 and 14 the results are presented with more detail and with the calculation of the relatives errors of all the implemented scheme.

x	Ref	**CDS**		**UDS**		**Hybrid**		**Power-law**	
0	1.989	1.849	7.02%	1.849	7.09 %	1.849	7.02%	1.849	7.02%
0.1	1.402	1.384	1.26%	1.379	1.64%	1.384	1.26%	1.384	1.26%
0.2	1.146	1.136	0.85%	1.129	1.44%	1.136	0.84%	1.136	0.86%
0.3	0.946	0.940	0.61%	0.932	1.41%	0.940	0.61%	0.940	0.63%
0.4	0.775	0.770	0.55%	0.763	1.54%	0.771	0.55%	0.771	0.57%
0.5	0.621	0.618	0.44%	0.611	1.61%	0.618	0.44 %	0.618	0.46%
0.6	0.480	0.478	0.37%	0.472	1.70%	0.478	0.37%	0.478	0.40%
0.7	0.349	0.348	0.22%	0.343	1.70%	0.348	0.22%	0.348	0.26%
0.8	0.227	0.227	0.20%	0.223	1.78%	0.227	0.20%	0.227	0.24%
0.9	0.111	0.111	0.35%	0.110	1.31%	0.111	0.35%	0.111	0.31%
1.0	0.000	0.000	-	0.000	-	0.000	-	0.000	-

Table 12: Results for $\rho/\Gamma = 10$

x	Ref	**CDS**		**UDS**		**Hybrid**		**Power-law**	
0	2.000	2.000	0.00%	2.000	0.00%	2.000	0.00%	2.000	0.00%
0.1	1.999	2.000	0.05%	1.999	0.00%	2.000	0.05%	2.000	0.05%
0.2	1.9997	1.999	0.005%	1.997	0.02%	1.999	0.02%	1.999	0.03%
0.3	1.985	1.992	0.34%	1.941	0.22%	1.974	0.56%	1.970	0.75%
0.4	1.841	1.824	0.91%	1.611	1.25%	1.698	7.75%	1.688	8.34%
0.5	0.951	0.967	1.65%	0.897	5.60%	0.907	4.65%	0.908	4.54%
0.6	0.154	0.149	3.42%	0.273	7.70%	0.220	42.6%	0.228	4.78%
0.7	0.001	0.006	503.2%	0.040	3880.3%	0.021	1992.5%	0.023	2206.3%
0.8	0.000	0.000	-	0.003	-	0.000	-	0.000	-
0.9	0.000	0.000	-	0.000	-	0.000	-	0.000	-
1.0	0.000	0.000	-	0.000	-	0.000	-	0.000	-

Table 13: Results for $\rho/\Gamma = 10^3$

The table enrich what it was discussed with the figures. The relative error calculation is not possible when the reference values are 0. Moreover, the relative errors are really high at $x = 0.7$ due to the fact that the correct values are really small and not accurately predicted by any of the schemes.

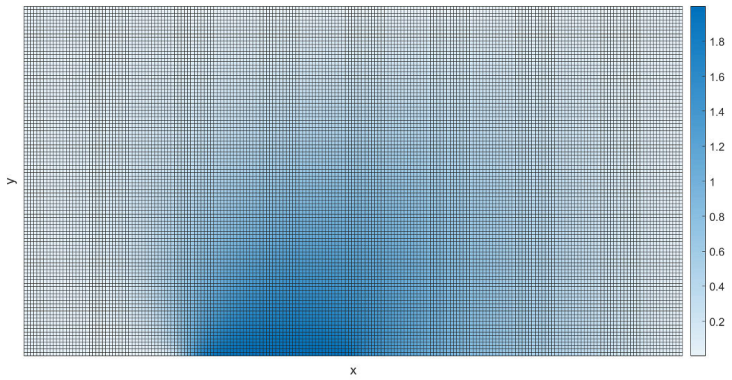
4.5.4. Results

Figure 29 shows the property Φ map for the three cases of the problem. These maps were obtained using a CDS scheme. It can be noted that as ρ/Γ increases, the map turns more symmetric. The Smith-Hutton problem is an excellent test of a numerical convection-diffusion scheme, especially in the high-convection regime [7]. Moreover, the transported variable

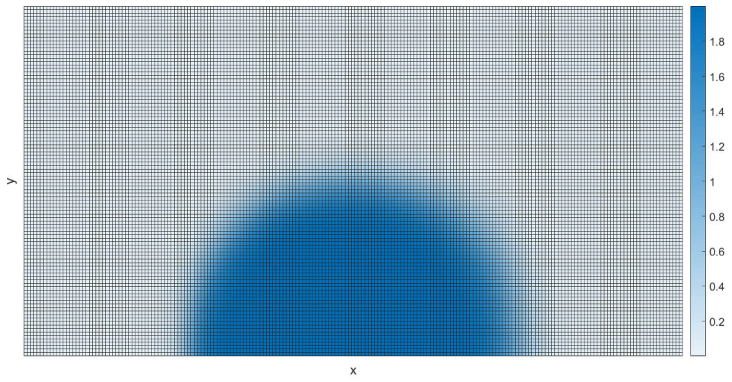
x	Ref	CDS		UDS		Hybrid		Power-law	
0	2.000	2.000	0.00%	2.000	0.00%	2.000	0.00%	2.000	0.00%
0.1	2.000	2.000	0.00%	2.000	0.00%	2.000	0.00%	2.000	0.00%
0.2	2.000	2.000	0.00%	2.000	0.02%	2.000	0.02%	1.999	0.02%
0.3	1.999	1.999	0.01%	1.980	0.96%	1.980	0.96%	1.980	0.96%
0.4	1.964	1.963	0.02%	1.717	12.6%	1.717	12.6%	1.717	12.6%
0.5	1.000	1.000	0.02%	0.904	9.60%	0.904	9.60%	0.904	9.60%
0.6	0.036	0.036	0.9%	0.205	470.5%	0.205	470.3%	0.205	470.3%
0.7	0.001	0.001	30.8%	0.017	1631.6 %	0.017	1629.7%	0.017	1626.7%
0.8	0.000	0.000	-	0.000	-	0.000	-	0.000	-
0.9	0.000	0.000	-	0.00	-	0.000	-	0.000	-
1.0	0.000	0.000	-	0.00	-	0.000	-	0.000	-

Table 14: Results for $\rho/\Gamma = 10^6$

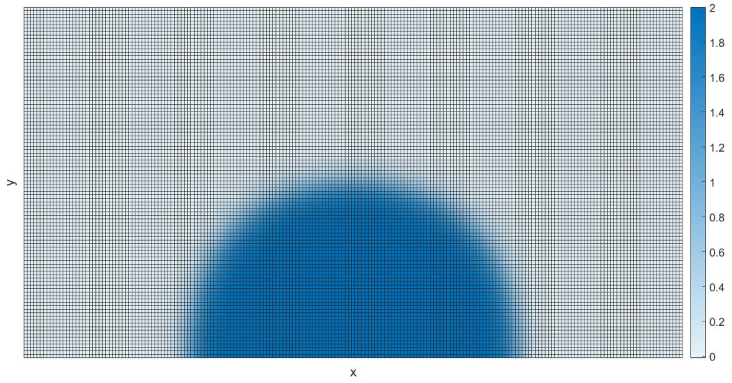
displays a rapid change over a small distance within the flow, simulating the a consequence of a source, the mixing of two streams of different temperature, etc.



(a) Property Φ *map for* $\rho/\Gamma = 10$



(b) Property Φ *map for* $\rho/\Gamma = 10$



(c) Property Φ *map for* $\rho/\Gamma = 10$

Figure 29: Property Φ *map with different Peclet numbers*

UDS is characterized by false diffusion, which can be noticed at high Peclet numbers. The scheme, anyway, gives physically plausible results. Exponential-based schemes such as Hybrid and Power-law, all revert to first-order upwinding and they present the same problem. CDS tends to give unrealistic results when the mesh refinement is insufficient but, if the mesh is refined, an accurate solution will result. There does not appear to be a perfect scheme. On regular meshes, all the methods submitted showed evidence of false diffusion.

5. Incompressible flow method using the Navier-Stokes equations

This section shows how the Fractional Step Method (FSM) [11] is applied to incompressible flows of Newtonian fluids in order to solve the velocity-pressure coupling in the Navier-Stokes Equations (NSE). The fractional step method (FSM) provides an approach that does not use pressure in a predictor step to estimate the projected velocity as an approximate solution of the momentum equation (like in the SIMPLE algorithm). Consequently, it is required to couple a pressure equation which determines the minimum perturbation that will make the predictor velocity incompressible. It is also important to recall that the role of the pressure in an incompressible flow is to enforce continuity; in some sense, it is more a mathematical variable than a physical one. [12]. There are two main goals of this chapter; the first is to depict the FSM employing staggered meshes for its solution, and the second is to assess the performance the numerical schemes presented in Section 4.2 in the solution of tow model problems, the Lid-Driven Cavity flow problem and the Differentially Heated Cavity flow problem.

5.1. Introduction to the Fractional Step Method

The FSM has become a very popular technoque for solving the incompressible NSE. The main reasons of its popularity is due to the fact that it has better performance than other methods such as SIMPLE-like algorithm [8][2]. Fractional step methods are also referred to as projection methods because the system of equations given can be interpreted as a projection into a divergence-free velocity space. The predictor velocity, is an approximate solution of the momentum equations, but because the predictor velocity is obtained with no pressure gradient contribution it cannot satisfy the incompressibility constraint at the next time level. Therefore, the Poisson equation determines the minimum perturbation that will make the predictor velocity incompressible [14]. Under the assumptions of incompressible flows of Newtonian fluids, the dimensionless governing equations in primitive variables are:

$$\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla)\mathbf{u} = \frac{1}{Re}\Delta \mathbf{u} - \nabla p \tag{5.1.1}$$

$$\nabla \cdot \mathbf{u} = 0 \tag{5.1.2}$$

where Re is the dimensionless Reynolds number defined as:

$$Re = \frac{\rho V_0 L}{\mu} \tag{5.1.3}$$

where ρ and μ are the density and the dynamic viscosity of the working fluid and L and V_0 are the characteristic length and velocity, respectively.

5.1.1. Application of the Helmholtz-Hodge decomposition theorem

In this section, the Helmholtz-Hodge decomposition theorem is applied to the incompressible NSE [14]. According to the Helmholtz-Hodge theorem, a given vector field ω, defined in a bounded domain Ω with smooth boundary $\delta\Omega$, is uniquely decomposed in a pure gradient field and a divergence-free vector parallel to $\delta\Omega$.

$$\omega = \mathbf{a} + \nabla\phi \tag{5.1.4}$$

where,

$$\nabla \cdot \mathbf{a} = 0 \quad \mathbf{a} \in \omega \tag{5.1.5}$$

Let $\Pi(\cdot)$ be a projector operator. It projects any vector field onto a divergence-free space

$$\nabla \cdot \Pi(\mathbf{a}) = 0 \tag{5.1.6}$$

Now, taking the NSE and applying the projector operator

$$\Pi\left(\frac{\partial \mathbf{u}}{\partial t} + \nabla p\right) = \Pi\left(-(\mathbf{u}\cdot\nabla)\mathbf{u} + \frac{1}{Re}\Delta\mathbf{u}\right) \tag{5.1.7}$$

Since the velocity field is incompressible for hypothesis, the transient term remains unchanged when projected:

$$\Pi\left(\frac{\partial \mathbf{u}}{\partial t}\right) = \frac{\partial \mathbf{u}}{\partial t} \tag{5.1.8}$$

While the projection of the pressure gradient vanishes

$$\Pi(\nabla p) = 0 \tag{5.1.9}$$

Therefore, the NS equations can be split in two different parts: a divergence-free vector and a gradient of a scalar field.

$$\frac{\partial \mathbf{u}}{\partial t} = \Pi\left(-(\mathbf{u}\cdot\nabla)\mathbf{u} + \frac{1}{Re}\Delta\mathbf{u}\right) \tag{5.1.10}$$

$$\nabla p = -(\mathbf{u}\cdot\nabla)\mathbf{u} + \frac{1}{Re}\Delta\mathbf{u} - \Pi\left(-(\mathbf{u}\cdot\nabla)\mathbf{u} + \frac{1}{Re}\Delta\mathbf{u}\right) \tag{5.1.11}$$

The Helmholtz-Hodge decomposition theorem ensures that this decomposition is unique. Finally, applying the divergence operator to 5.1.11 and using thr projector definition (Eq. 5.1.6), leads to the Poisson equation for the pressure

$$\nabla p = \nabla \cdot \left(-(\mathbf{u}\cdot\nabla)\mathbf{u} + \frac{1}{Re}\Delta\mathbf{u}\right) \tag{5.1.12}$$

If $R(\mathbf{u})$ stands for the convective and diffusive terms:

$$R(\mathbf{u}) \equiv -(\mathbf{u}\cdot\nabla)\mathbf{u} + \frac{1}{Re}\Delta\mathbf{u} \tag{5.1.13}$$

For incompressible flows, the role of the pressure gradient is to project the vector field $R(\mathbf{u})$ into the divergence-free space.

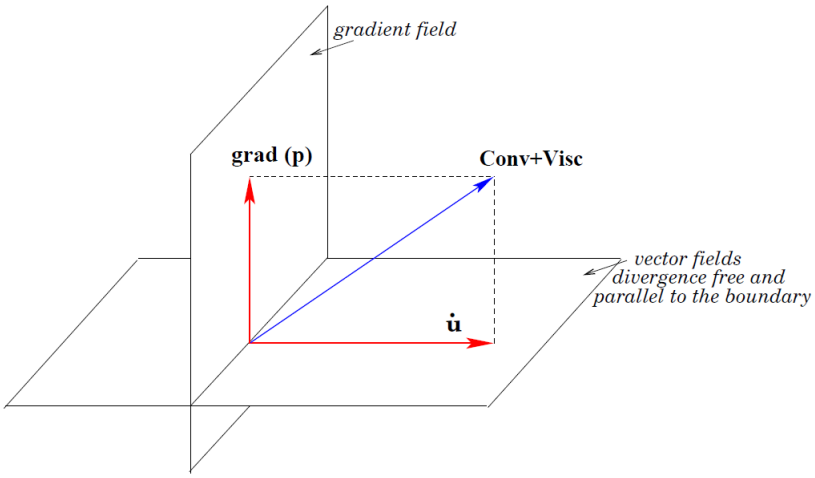


Figure 30: Convective and viscous term vector field decomposition [14]

5.1.2. Time-integration method

The final form of the fractional step method would depend on the time-integration method chosen. Here, for the sake of clarity, it is proposed a fully explicit time integration scheme. To simplify the notation, momentum equation can be rewritten as

$$\frac{\partial\mathbf{u}}{\partial t} = R(\mathbf{u}) - \nabla p \tag{5.1.14}$$

For the temporal discretization, a central difference scheme is used for the time derivative term

$$\frac{\partial\mathbf{u}}{\partial t}\Big|^{n+1/2} \approx \frac{\mathbf{u}^{n+1}-\mathbf{u}^{n}}{\Delta t} + O(\Delta t^2) \tag{5.1.15}$$

a fully explicit second-order Adams-Bashfort scheme for $R(\mathbf{u})$

$$R^{n+1/2}(\mathbf{u}) \approx \frac{3}{2}R(\mathbf{u}^n) - \frac{1}{2}R(\mathbf{u}^{n-1}) + O(\Delta t^2, \Delta x^m) \tag{5.1.16}$$

and a first order backward Euler scheme for the pressure-gradient term. Incompressiblity constraint is treated implicitly. Thus, the following semi-descritized NSE is obtained

$$\frac{\mathbf{u}^{n+1}-\mathbf{u}^{n}}{\Delta t} = \frac{3}{2}R(\mathbf{u}^n) - \frac{1}{2}R(\mathbf{u}^{n-1}) - \nabla p^{n+1} \tag{5.1.17}$$

$$\nabla \cdot \mathbf{u}^{n+1} = 0 \tag{5.1.18}$$

It can be noted that the pressure gradient forces (projects) the predictor velocity field to be incompressible ($\nabla \cdot \mathbf{u}^{n+1} = 0$). This projection is derived from the well-known Helmholtz-Hodge vector decomposition theorem, whereby the predictor velocity $\mathbf{u}^p$, can be uniquely decomposed into a divergence-free vector, $\mathbf{u}^{n+1}$, and the gradient of a scalar field, $\nabla\widetilde{p}$. This decomposition is written as

$$\mathbf{u}^p = \mathbf{u}^{n+1} + \nabla\widetilde{p} \tag{5.1.19}$$

where the predictor velocity vector $\mathbf{u}^p$ is given by

$$\mathbf{u}^p = \mathbf{u}^n + \Delta t \left(\frac{3}{2} R(\mathbf{u}^n) - \frac{1}{2} R(\mathbf{u}^{n-1}) \right) \tag{5.1.20}$$

and the pseudo-pressure is $\widetilde{p} = \frac{\Delta t}{\rho} p^{n+1}$. Taking the divergence of Eq. (5.1.19) yields a Poisson equation for $\widetilde{p}$.

$$\nabla \cdot \mathbf{u}^p = \nabla \cdot \mathbf{u}^{n+1} + \nabla \cdot (\nabla\widetilde{p}) \rightarrow \Delta\widetilde{p} = \nabla \cdot \mathbf{u}^p \tag{5.1.21}$$

Once the solution is obtained, $\mathbf{u}^{n+1}$ results from the correction

$$\mathbf{u}^{n+1} = \mathbf{u}^p - \Delta\widetilde{p} \tag{5.1.22}$$

Therefore, the algorithm for the integration of each time step is

1. Evaluate $R(\mathbf{u}^n)$.
2. Evaluate $\mathbf{u}^p$ from Eq. (5.1.20.)
3. Evaluate $\nabla \cdot \mathbf{u}^p$ and solve the discrete Poisson equation, Eq. (5.1.21).
4. Obtain the new velocity field with Eq.(5.1.22)

5.1.3. Determination of Δt

Due to stability reasons, explicit temporal schemes introduce severe restrictions on the time step, while implicit discretization would improve the overall stability. Therefore, having considered an explicit method in the view of formulation simplicity, the time-step Δt must be bounded by the CFL condition [13] given by

$$\Delta t \left(\frac{|\mathbf{u}_i|}{\Delta x_i} \right)_{max} \leq C_{conv} \tag{5.1.23}$$

$$\Delta t \left(\frac{\nu}{\Delta x_i^2} \right)_{max} \leq C_{visc} \tag{5.1.24}$$

where the bounding values C_{conv} and C_{visc} must be smaller than unity. In this case, the recommendations given by [15] are adopted, using values $C_{conv} = 0.35$ and $C_{visc} = 0.2$, respectively.

5.1.4. Solution of Poisson Equation

Since the formulation is fully explicit, the only system to be solved is the pressure Poisson Eq.(5.1.21). Therefore, the efficient resolution of this equation is of critical importance. When the velocity field is computed through the predictor velocities, it is necessary to calculate the pressure gradient at each node. Supposing a 1D situation, and isolating the velocity at instant $t = n+1$ from Eq.(5.1.19), it yields to the following equation

$$u_P^{n+1} = u_P^p - \frac{\Delta t}{\rho}\left(\frac{p_E^{n+1} - p_W^{n+1}}{2\Delta x}\right) \tag{5.1.25}$$

Having considered the 1D spacial grid of figure 31.

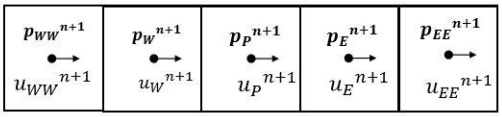


Figure 31: 1D spatial discretization [16]

The problem in equation 5.1.19 is that the gradient of pressure at the node P is independent of the pressure at node P and it can generate a nonphysical pressure distribution even if the velocity field has reached convergence. For example, considering the values

$$p_{WW} = 100, \quad p_W = 0, \quad p_P = 100, \quad p_E = 0, \quad p_{EE} = 100.$$

it can be observed that verifies $\nabla p^{n+1} = 0$ although it is a nonphysical solution. A possible solution to this problem is to use staggered meshes. In this way, it is possible to avoid the checkerboard problem.

5.1.5. Staggered Meshes

A displaced or "staggered" grid for the velocity components was first used by Harlow and Welch in [2]. In the staggered grid, the velocity components are calculated for the points that lie on the faces of the control volumes. Figure 32 is a basic example of a staggered mesh.

The main mesh is only used to get the pressure field. Then, it is easy to see that the nodes which belong to the main mesh are located at the cell faces of the staggered meshes. It enables to compute the pressure gradient at the determined node of the staggered mesh, since it is exactly in the middle between the two main nodes of the mesh. Doing this, checkerboard

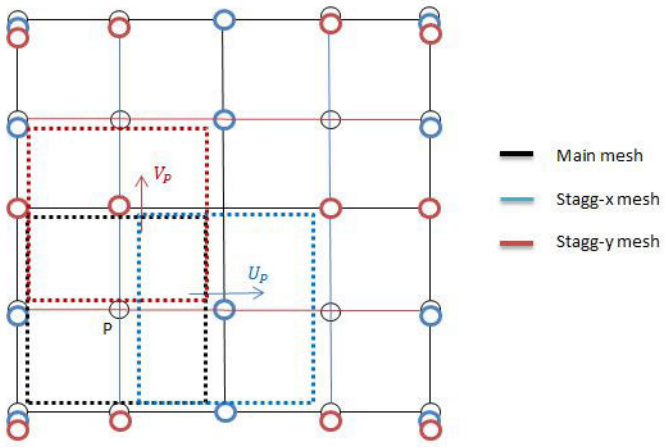


Figure 32: Example of a staggered mesh [9]

problem is avoided. However, it means that all the 'x' components of the velocity will be computed in the staggered 'x' mesh, and the 'y' components in the staggered 'y' mesh.

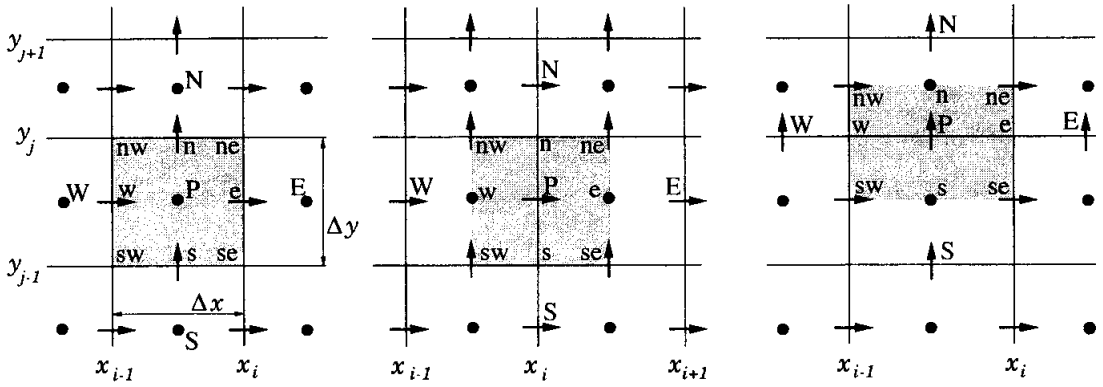


Figure 33: Control volumes for a staggered grid: for mass conservation and scalar quantities (left), for x-momentum (center) and for y-momentum (right). Extracted from [12]

Figure 33 shows that the control volumes for u and v, are displaced with respect to the control volume for the continuity equation. The advantages of implementing a staggered grid have their own prices. A computer program based on a staggered grid must carry all the indexing and geometric information about the locations of the velocity components and some tiresome interpolations are needed. In any case, the benefits of the staggered grid are worth the additional complexity.

5.2. The Lid-Driven Cavity flow problem

The Driven Cavity Flow problem is a model problem that has served repeatedly for testing and evaluating numerical techniques. The results of this problem, for moderately high Reynolds (Re) number values, have been published in a number of sources, using a variety

of solution procedures, including an attempt to analytically extract the corner singularities from the dependant variable of the problem [17]. Some accurate results are now available for high Re also [17] and they will be used for validating the resolution methods explained in this work.

5.2.1. Problem definition

Laminar incompressible flow in a square cavity whose top wall moves with a uniform velocity in its own plane is considered. A simple scheme is shown in figure 34.

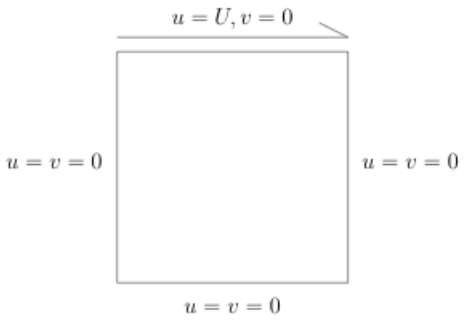


Figure 34: Scheme of Lid-Driven Cavity problem [18]

The Reynolds number is an input of the problem, since it determines the performance of the fluid. Velocities u and v are zero in the right, left and bottom walls; in the top side, the velocity is horizontal and equals to 1 m/s. The boundary conditions are that the pressure gradient normal to the walls is zero. The cavity is modeled as a 1x1 m^2 square. It is required to solve the velocity field (x and y directions), in particular, numerical values corresponding to the velocity profiles for horizontal and vertical lines passing through the geometric center of the cavity. Cases to be assessed are $Re = 100, 400, 1000, 3200$.

5.2.2. Code development

Problem assumptions

1. Pseudo-transient simulation with adaptive time step;
2. Two-dimensional problem, the third dimension is taken as unity;
3. The cavity is square, with $L = 1\ m$ and $H = 1\ m$;
4. FVM, cell-centered discretization: average values over each cell;
5. Constant thermophysical properties: density ρ and viscosity μ;
6. All the velocity at the faces are calculated according to a convective scheme.

Mesh generation

The FSM method is applied along with uniform meshes. Three meshes are created as shown in Fig. 32 and Fig. 33: the main mesh is used for pressure while the additional two staggered meshes are created for the calculations of the velocity fields. The implemented mesh is shown in Fig. 35.

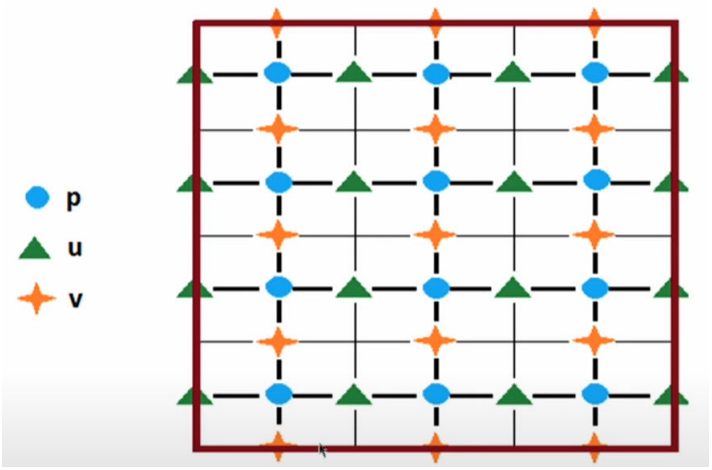


Figure 35: Pressure, staggered-x and staggered-y meshes [19]

Code equations

At each time step, the following equations give a unique $\mathbf{u}^{n+1}$ and p^{n+1}.

1. $\mathbf{u}^p = \mathbf{u}^n + \Delta t \left(\frac{3}{2}R(\mathbf{u}^n) - \frac{1}{2}R(\mathbf{u}^{n-1})\right)$
2. $\Delta\widetilde{p} = \nabla \cdot \mathbf{u}^p$
3. $\mathbf{u}^{n+1} = \mathbf{u}^p - \Delta\widetilde{p}$
4. $\Delta t^{n+1} = min(\Delta t_c, \Delta t_d)$

These equations have to be discretized in the different components of the velocity field.

1 - x component of v

The x component of the predicted velocity is calculated as:

$$u^p = u^n + \left[\frac{3}{2}R(u^n) - \frac{1}{2}R(u^{n-1})\right] \tag{5.2.1}$$

Where:

$$R(u) = -(v \cdot \nabla)u + \frac{1}{Re}\Delta u \tag{5.2.2}$$

If $R(u)$ is integrated over the staggered-x control volume and then the divergence theorem is applied

$$\int_{V_{cv}} R(u)\, dV_{cv} = -\int_{V_{cv}} (v\cdot\nabla)u\, dV_{cv} + \int_{V_{cv}} \Delta u\, dV_{cv} = -\int_{S_{cv}} (v\cdot\nabla)u\cdot n\, dS_{cv} + \int_{S_{cv}} \nabla u\cdot n\, dS_{cv} \tag{5.2.3}$$

Introducing the second-order approximations

$$R(u)V_{cv} = [-(u)_e u_e A_e + (u)_w u_w A_w - (v)_n u_n A_n + (v)_s u_s A_s] + \frac{1}{Re}\left[\frac{u_E - u_P}{d_{EP}} - \frac{u_P - u_W}{d_{WP}} + \frac{u_N - u_P}{d_{NP}} - \frac{u_P - u_S}{d_{SP}}\right] \tag{5.2.4}$$

As said before, the convective schemes are used to calculate the velocities at the faces $(u)_f$ and $(v)_f$ and linear interpolation is used to estimate the terms $(u)_f$ and $(v)_f$; an example is shown in Fig.36.

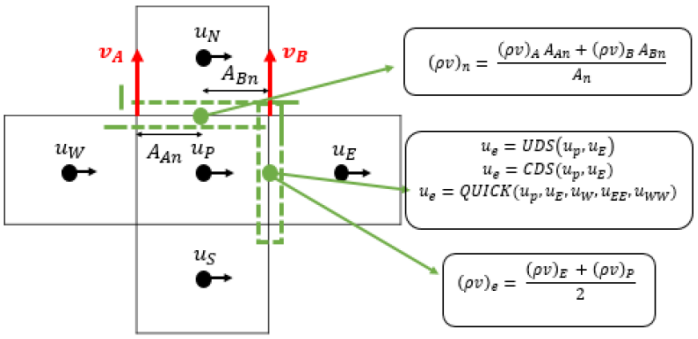


Figure 36: Approximation of the velocities at the faces. [CTTC,2017]

1 - y component of v

The same procedure is applied to the y component of the predicted velocity.

$$v^p = v^n + \left[\frac{3}{2}R(v^n) - \frac{1}{2}R(v^{n-1})\right] \tag{5.2.5}$$

Where

$$R(v) = -(v\cdot\nabla)v + \frac{1}{Re}\Delta v \tag{5.2.6}$$

Integrating again $R(v)$ over the staggered-y control volume and applying the divergence theorem.

$$\int_{V_{cv}} R(v)\, dV_{cv} = -\int_{V_{cv}} (v\cdot\nabla)v\, dV_{cv} + \int_{V_{cv}} \Delta v\, dV_{cv} = -\int_{S_{cv}} (v\cdot\nabla)v\cdot n\, dS_{cv} + \int_{S_{cv}} \nabla v\cdot n\, dS_{cv} \tag{5.2.7}$$

Introducing the second-order approximations

$$R(u)V_{cv} = [-(u)_e v_e A_e + (u)_w v_w A_w - (v)_n v_n A_n + (v)_s v_s A_s] + \frac{1}{Re}\left[\frac{v_E - v_P}{d_{EP}} - \frac{v_P - v_W}{d_{WP}} + \frac{v_N - v_P}{d_{NP}} - \frac{v_P - v_S}{d_{SP}}\right] \tag{5.2.8}$$

2 - Poisson equations

Integrating over the main mesh control volume and applying the divergence theorem

$$\int_{V_{cv}} \Delta\widetilde{p}^{n+1}\, dV_{cv} = \int_{V_{cv}} \nabla \cdot \mathbf{u}^p\, dV_{cv} \tag{5.2.9}$$

$$\int_{S_{cv}} \nabla\widetilde{p}^{n+1} \cdot n\, dS_{cv} = \int_{S_{cv}} \mathbf{u}^p \cdot n\, dS_{cv} \tag{5.2.10}$$

And introducing the second-order approximations

$$\frac{p_E^{n+1} - p_P^{n+1}}{d_{EP}} A_e - \frac{p_P^{n+1} - p_W^{n+1}}{d_{WP}} A_w + \frac{p_N^{n+1} - p_P^{n+1}}{d_{NP}} A_n - \frac{p_P^{n+1} - p_S^{n+1}}{d_{SP}} A_s = \frac{1}{\Delta t}[(u^p)_e A_e - (u^p)_w A_w + (v^p)_n A_n - (v^p)_s A_s] \tag{5.2.11}$$

Rearranging in the usual discretization equation

$$a_P p_P^{n+1} = a_E p_E^{n+1} + a_W p_W^{n+1} + a_N p_N^{n+1} + a_S p_S^{n+1} + b_P \tag{5.2.12}$$

Where

$$a_E = \frac{A_e}{d_{EP}}$$

$$a_W = \frac{A_w}{d_{WP}}$$

$$a_N = \frac{A_n}{d_{NP}}$$

$$a_s = \frac{A_s}{d_{SP}}$$

$$a_P = a_E + a_W + a_N + a_S$$

$$b_P = -\frac{1}{\Delta t}[(u^p)_e A_e - (u^p)_w A_w + (v^p)_n A_n - (v^p)_s A_s]$$

The GS solver is used in order to solve this set of equations.

Boundary conditions

Two types of boundary conditions are applied to the problem.

1. Prescribed velocity

   Referring to Fig.34 and considering the system of coordinate origin in the lower left

corner

$$u(x,y) = U \ \ on \ \ H = 1 \tag{5.2.13}$$

$$u(x,y) = v(x,y) = 0 \ \ \ elsewhere \tag{5.2.14}$$

2. Pressure gradient
   The pressure gradient is zero along all the borders.

$$\frac{\partial p}{\partial n} = 0 \ \ \ everywhere \tag{5.2.15}$$

3 - Updated velocity field on staggered - x mesh

The new velocity field can finally be calculated. Referring to Fig. 37 the x component can be calculated as:

$$u_P^{n+1} = u_P^p - \left(\frac{\partial \widetilde{p}}{\partial x}\right)_P^{n+1} \tag{5.2.16}$$

$$u_P^{n+1} = u_P^p - \frac{\widetilde{p}_B^{n+1} - \widetilde{p}_A^{n+1}}{d_{BA}}$$

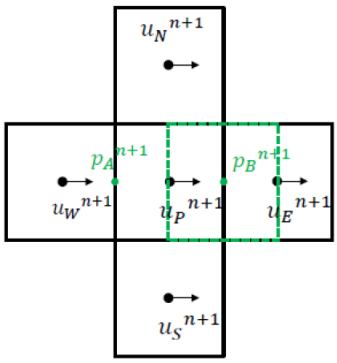


Figure 37: Geometric relation of the main mesh and the x-staggered mesh.[CTTC,2017]

It can be noted that for the velocities u_P^{n+1} of the x-staggered mesh, $p^{n+1}(i, j+1)$ and $p^{n+1}(i, j)$ are the correspondent pressure for p_B^{n+1} and p_A^{n+1} respectively; another advantage for using staggered meshes.

3 - Updated velocity field on staggered - y mesh

Similar calculations are performed on the y component of the velocity field.

$$v_P^{n+1} = v_P^p - \left(\frac{\partial \widetilde{p}}{\partial y}\right)_P^{n+1} \tag{5.2.17}$$

$$v_P^{n+1} = v_P^p - \frac{\widetilde{p}_B^{n+1} - \widetilde{p}_A^{n+1}}{d_{BA}}$$

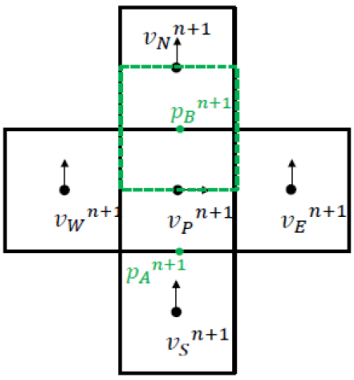


Figure 38: Geometric relation of the main mesh and the y-staggered mesh.[CTTC,2017]

Again, it can be noted that for the velocities v_P^{n+1} of the y-staggered mesh, $p^{n+1}(i+1,j)$ and $p^{n+1}(i,j)$ are the correspondent pressure for p_B^{n+1} and p_A^{n+1} respectively.

4 - Choice of the new time step

The CFL (Courant-Friedrich-Levy) condition has to be respected in order to maintain numerical stability.

$$\Delta t_c = min\left(0.35\frac{\Delta x}{|v|}\right) \tag{5.2.18}$$

$$\Delta t_d = min\left(0.20\frac{\rho\Delta x^2}{\mu}\right) \tag{5.2.19}$$

$$\Delta t = min(\Delta t_c, \Delta t_d) \tag{5.2.20}$$

This equations are valid only for a uniform mesh where $\Delta x = \Delta y$

5.2.3. Description of the algorithm

The algorithm proposal for the resolution of the problem is presented in Fig.39 and explained below.

1. Input Data

   - Physical data: Reynolds number and the square domain dimensions;
   - Numerical data: number of control volumes, convective scheme and convergence criteria (δ);
   - Boundary conditions: $u, v = 0$ at boundaries, except $u = U$ at the upper boundary;

2. Mesh generation.
   It is necessary to compute the 3 different meshes as explained before. Each mesh has to have the following data: position of the nodes, areas of the cell faces and volumes;

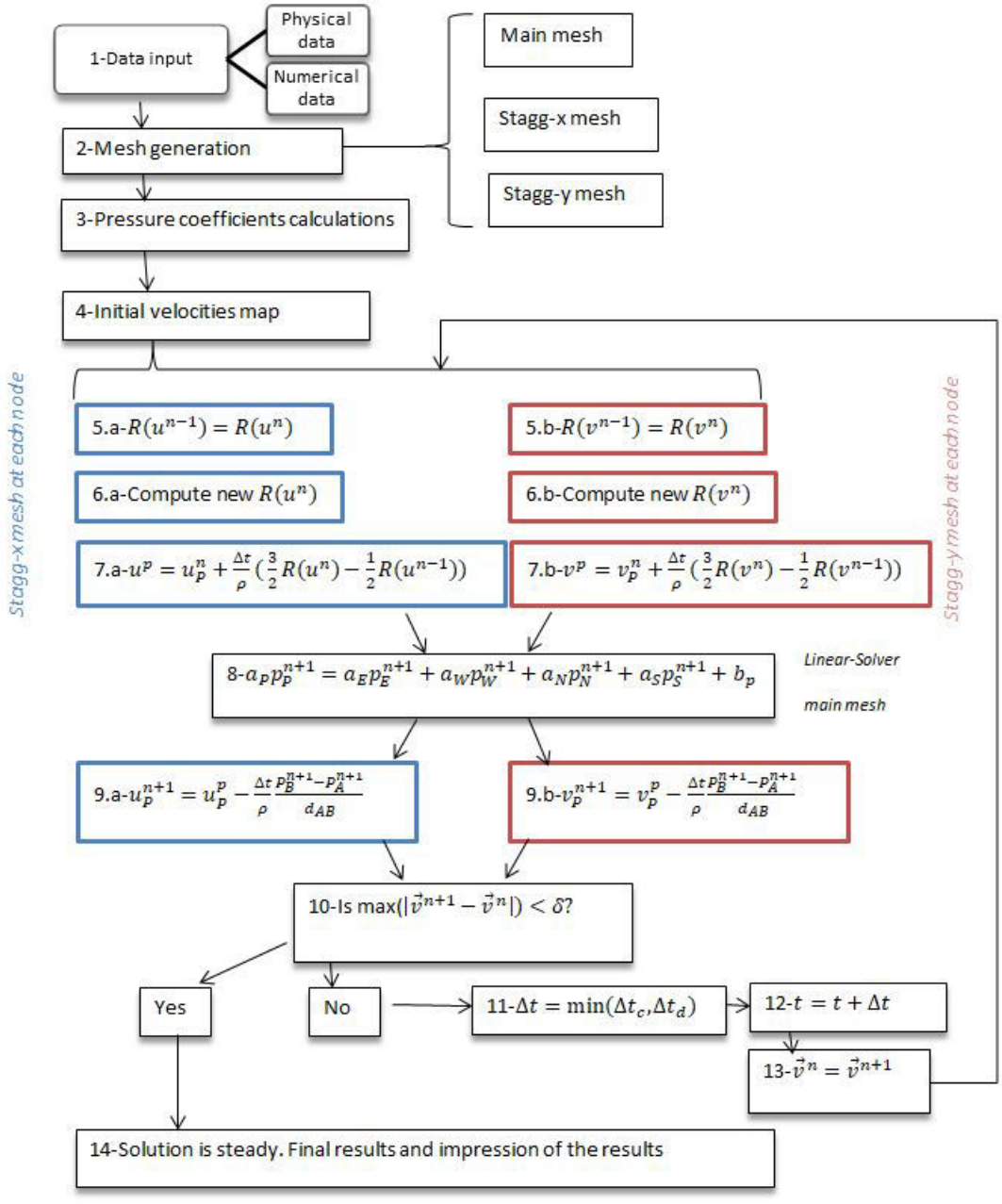


Figure 39: Fractional Step Method code scheme [9]

3. Initial calculations
   The values of the initial velocities for each staggered mesh have to be set: $u^n[i,j]$, $v^n[i,j]$, $u^{n+1}[i,j]$, $v^{n+1}[i,j]$. Moreover, the first time is computed as: $\Delta t = min(\Delta t_c, \Delta t_d)$

4. Calculation of the vector field $R(\mathbf{u})$;
   The value of $R(\mathbf{u})$ is calculated for the current and previous instant, with the chosen convective scheme;

5. Calculation of the predicted velocities;

6. Computation of the pressure fiels.
   With the predictor velocities is possible to calculate the coefficients of the discretized Poisson equation, and the resultant linear equations are solved through a linear solver;

7. New velocity field computation.
   With the predictor velocities and the pressure field at time $'n+1'$ it is possible to get the new velocity field for each staggered mesh;

8. Steady state condition verification
   If the solution becomes steady, then the velocity field will not change enough and the iterations will stop. Otherwise, if the new velocity field changes more than the criteria established (δ), the iterations will continue with a new time step. In mathematical formulas, if

$$max(|u^{n+1}[i,j] - u^n[i,j]|) \leq \delta \quad AND \quad max(|v^{n+1}[i,j] - v^n[i,j]|) \leq \delta \tag{5.2.21}$$

   convergence is reached, otherwise the program keeps running;

9. Update of the velocity field.
   The velocity field is updated as follows

$$u^{n-1} \leftarrow u^n, \;\; u^n \leftarrow u^{n+1}$$

$$v^{n-1} \leftarrow v^n, \;\; v^n \leftarrow v^{n+1}$$

10. Calculation of the new time-step.
    If the solution is still unsteady, new time step has to be chosen following the criteria of Courant-Friedrich-Levy;

11. Print the results.
    If the solution is steady, get the final results.

129-grid pt. no.	y	Re						
		100	400	1000	3200	5000	7500	10,000
129	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
126	0.9766	0.84123	0.75837	0.65928	0.53236	0.48223	0.47244	0.47221
125	0.9688	0.78871	0.68439	0.57492	0.48296	0.46120	0.47048	0.47783
124	0.9609	0.73722	0.61756	0.51117	0.46547	0.45992	0.47323	0.48070
123	0.9531	0.68717	0.55892	0.46604	0.46101	0.46036	0.47167	0.47804
110	0.8516	0.23151	0.29093	0.33304	0.34682	0.33556	0.34228	0.34635
95	0.7344	0.00332	0.16256	0.18719	0.19791	0.20087	0.20591	0.20673
80	0.6172	−0.13641	0.02135	0.05702	0.07156	0.08183	0.08342	0.08344
65	0.5000	−0.20581	−0.11477	−0.06080	−0.04272	−0.03039	−0.03800	0.03111
59	0.4531	−0.21090	−0.17119	−0.10648	−0.86636	−0.07404	−0.07503	−0.07540
37	0.2813	−0.15662	−0.32726	−0.27805	−0.24427	−0.22855	−0.23176	−0.23186
23	0.1719	−0.10150	−0.24299	−0.38289	−0.34323	−0.33050	−0.32393	−0.32709
14	0.1016	−0.06434	−0.14612	−0.29730	−0.41933	−0.40435	−0.38324	−0.38000
10	0.0703	−0.04775	−0.10338	−0.22220	−0.37827	−0.43643	−0.43025	−0.41657
9	0.0625	−0.04192	−0.09266	−0.20196	−0.35344	−0.42901	−0.43590	−0.42537
8	0.0547	−0.03717	−0.08186	−0.18109	−0.32407	−0.41165	−0.43154	−0.42735
1	0.0000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000

Figure 40: Benchmark solution: u [17]

5.2.4. Verification

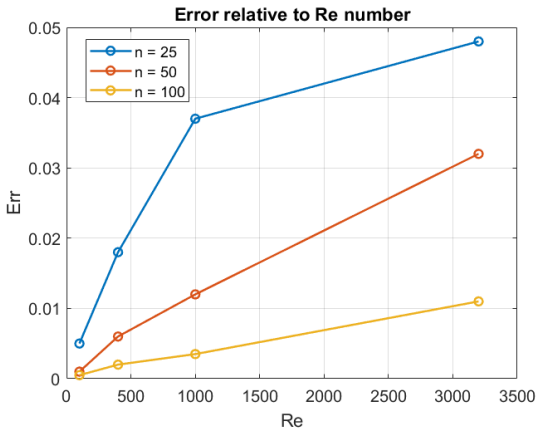
In order to verify the code, the results are compared with a benchmark solution extracted from literature [17], which are shown in tables 40 and 41 . The main code uses QUICK scheme, although it is compared with CDS scheme and UDS. However, results with QUICK are analysed for Reynolds below 5,000, since, as the flow gets turbulent, much finer meshes, which imply much higher computational costs, are needed (unless turbulence models are implemented, which is out of the scope of this work).

129-grid pt. no.	x	Re						
		100	400	1000	3200	5000	7500	10,000
129	1.0000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
125	0.9688	–0.05906	–0.12146	–0.21388	–0.39017	–0.49774	–0.53858	–0.54302
124	0.9609	–0.07391	–0.15663	–0.27669	–0.47425	–0.55069	–0.55216	–0.52987
123	0.9531	–0.08864	–0.19254	–0.33714	–0.52357	–0.55408	–0.52347	–0.49099
122	0.9453	–0.10313	–0.22847	–0.39188	–0.54053	–0.52876	–0.48590	–0.45863
117	0.9063	–0.16914	–0.23827	–0.51550	–0.44307	–0.41442	–0.41050	–0.41496
111	0.8594	–0.22445	–0.44993	–0.42665	–0.37401	–0.36214	–0.36213	–0.36737
104	0.8047	–0.24533	–0.38598	–0.31966	–0.31184	–0.30018	–0.30448	–0.30719
65	0.5000	0.05454	0.05186	0.02526	0.00999	0.00945	0.00824	0.00831
31	0.2344	0.17527	0.30174	0.32235	0.28188	0.27280	0.27348	0.27224
30	0.2266	0.17507	0.30203	0.33075	0.29030	0.28066	0.28117	0.28003
21	0.1563	0.16077	0.28124	0.37095	0.37119	0.35368	0.35060	0.35070
13	0.0938	0.12317	0.22965	0.32627	0.42768	0.42951	0.41824	0.41487
11	0.0781	0.10890	0.20920	0.30353	0.41906	0.43648	0.43564	0.43124
10	0.0703	0.10091	0.19713	0.29012	0.40917	0.43329	0.44030	0.43733
9	0.0625	0.09233	0.18360	0.27485	0.39560	0.42447	0.43979	0.43983
1	0.0000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000

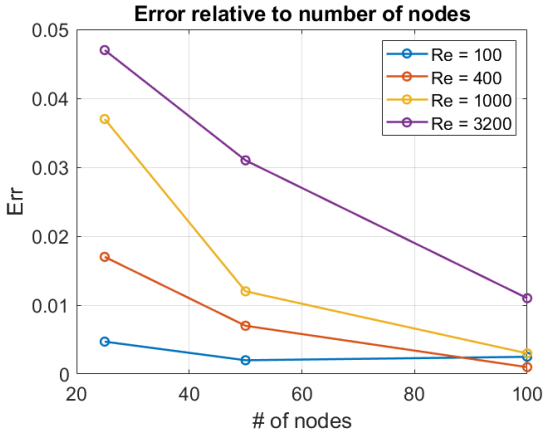
Figure 41: Benchmark solution: v [17]

A grid independence study is done in order to analyse the influence of the mesh on the results. Three meshes are employed (25,50 and 100 nodes) and the error is calculated for each mesh and each Reynolds number. Figure 43 represents the error of both the components of the velocity field respect to the benchmark solution. As it has been said, the solution is given only in determined points, so an interpolation has been performed in order to get an approximation of the velocity values at the determined coordinates of the solution, enabling to compare the velocities. Figure 43(a) and 43(b) shows the error between the horizontal velocities in the vertical central line. It can be noted that the error increases when increasing the Reynolds number; this behaviour is due to the fact that the flow becomes more turbulent and it is more difficult to achieve a steady solution without the use of a more refined mesh. It is also possible to observe that increasing the number of nodes decreases the error, that is what is expected. The same behaviour is also shown in figure 43(c) and 43(d), where the error of vertical velocities on the horizontal line is shown.

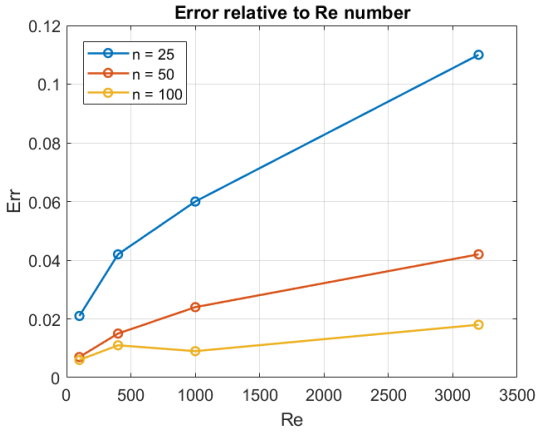
The second part of the study consists in comparing the benchmark solution from tables 40 and 41 with the results obtained through different meshes. This comparison is presented in Fig. 45, where the x component of the velocity along the vertical center line and the y component along the horizontal center line are shown for different Reynolds number. It can be noted that the points retrieved from the benchmark solution aren't enough to achieve an accurate shape in the graph, so it is important that the computes solution coincides with these points.



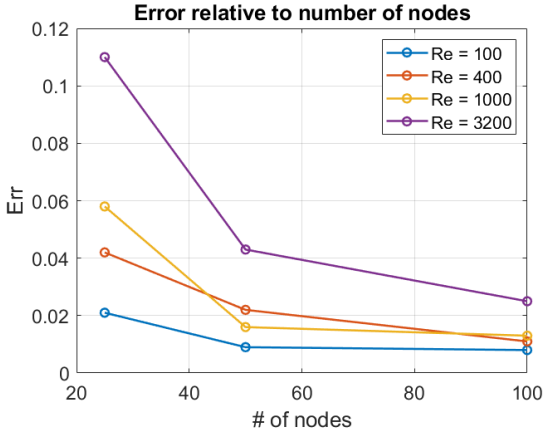
(a) Error vs Re number for u velocity components



(b) Error vs number of elements for u velocity components

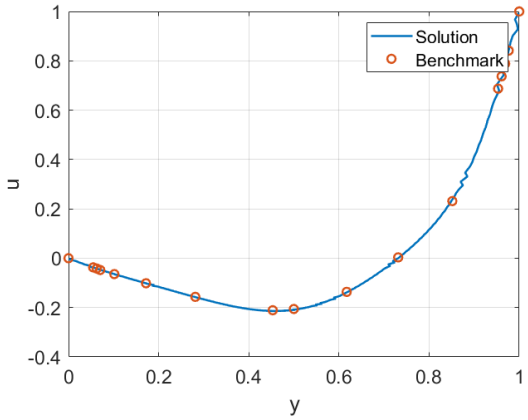


(c) Error vs Re number for v velocity components

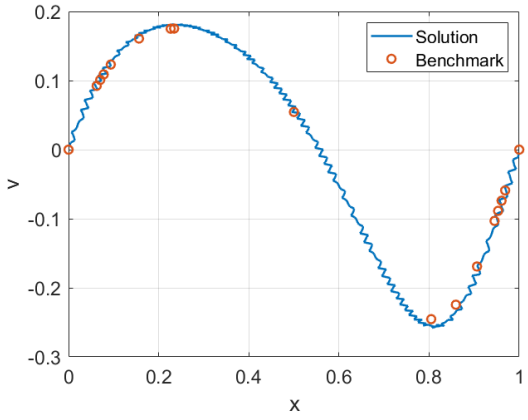


(d) Error vs number of elements for v velocity components

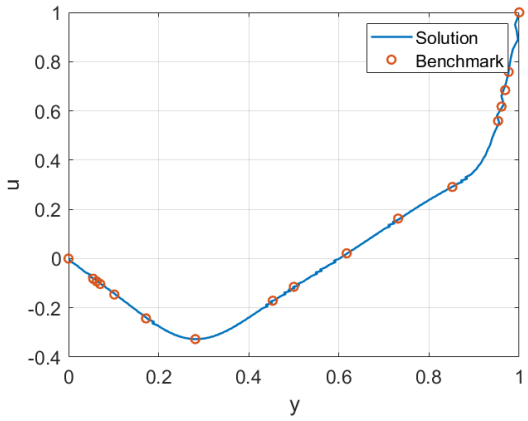
Figure 42: Relative errors compared with benchmark solution [17]



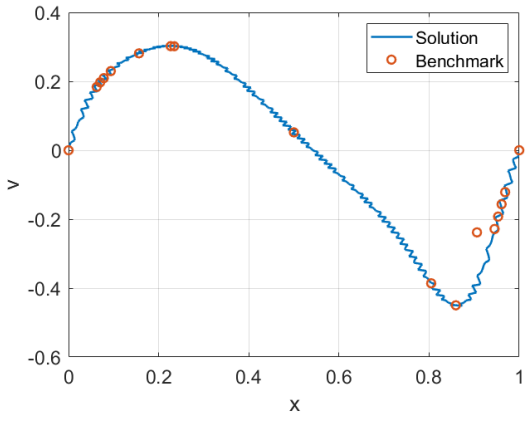
(a) u, $Re = 100$



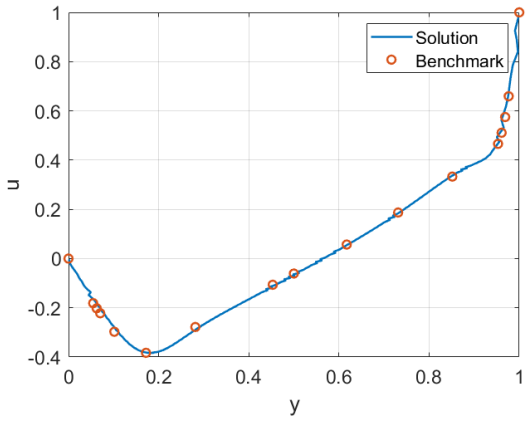
(b) v, $Re = 100$



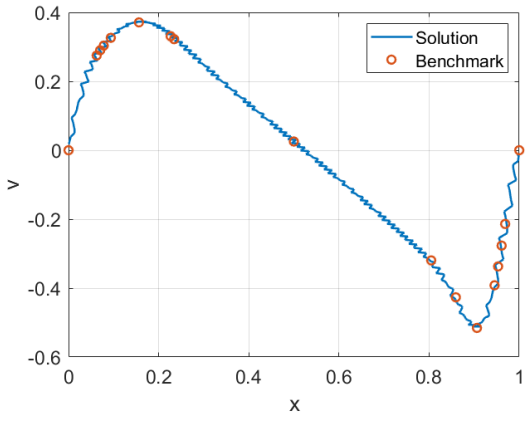
(c) u, $Re = 400$



(d) v, $Re = 400$

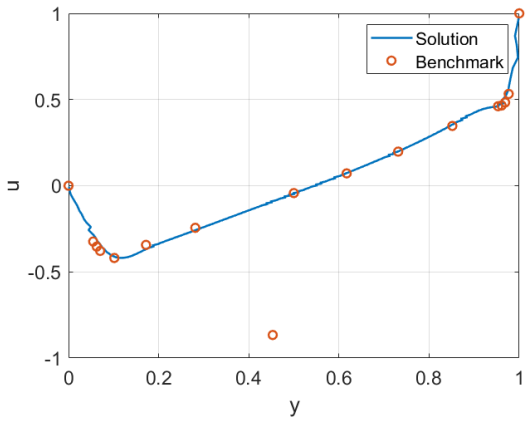


(e) u, $Re = 1000$

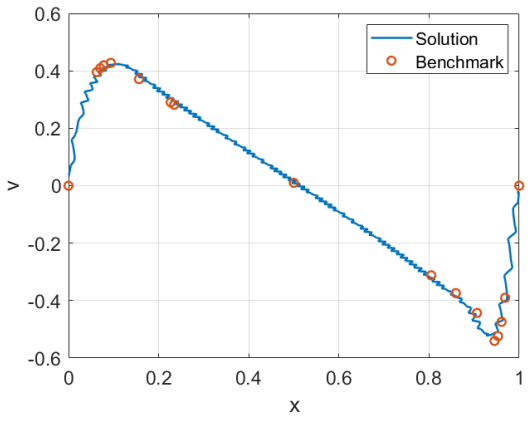


(f) v, $Re = 1000$

Figure 43: Lid-Driven Cavity results compared with benchmark solutions



(g) u, $Re = 3200$



(h) v, $Re = 3200$

Figure 43: Lid-Driven Cavity results compared with benchmark solutions

The obtained results seem to have a good accordance with the proposed benchmark values. Anyway, the results are not exactly coincident due to the coarse mesh involved, but they show the same trend and the same flow behaviour. It has to be noted that there is a point of the benchmark solution, for $Re = 400$ (Figure 43(d)), which really seem to divert to implemented solution.

All the results shown have been calculated, as said before, with the QUICK scheme. Anyway, the lower-order schemes UDS and UDS have been implemented to evaluate their performance. In Table 15 the average error relative to the benchmark solution is shown, for all the used meshes and schemes.

Mesh	**Reynolds**	**UDS**	**CDS**	**QUICK**
25x25	$Re = 100$	35.1%	5.3%	2.9%
	$Re = 400$	∞	10.1%	3.8%
	$Re = 1000$	∞	13.4%	6.5%
	$Re = 3200$	∞	∞	∞
50x50	$Re = 100$	9.7%	2.1%	1.5%
	$Re = 400$	∞	2.3%	1.8%
	$Re = 1000$	∞	3.7%	2.5%
	$Re = 3200$	∞	∞	∞
100x100	$Re = 1000$	-	-	1.1%
	$Re = 3200$	-	-	1.9%

Table 15: Lid-Driven Cavity average relative errors

"n.c" indicates a non-convergence of algorithm while for the most refined (100x100) only $Re = 1000$ and $Re = 3200$ were simulated, because good accuracy was already reached in the other simulations.

5.2.5. Final results

In the following pages contour plots of horizontal and vertical components of the velocity field and streamlines are shown.

In Figure 45 the contours plots of the horizontal and vertical component of velocity are displayed. The main remarkable aspect of these plots is that the areas of positive and negative velocity become more elongated with the increasing of *Re* number. This is due to the diffusive term being less predominant for higher *Re*. This phenomenon can be seen both for the horizontal and vertical velocities. In the end, it is clear observing Figure 45 the clockwise motion of the fluid inside the square cavity.

A stream function is a scalar function which is defined by the following two expressions [20]:

$$u = \frac{\partial \Psi}{\partial y} \tag{5.2.22}$$

$$v = -\frac{\partial \Psi}{\partial x} \tag{5.2.23}$$

Streamlines are a family of curves that are instantaneously tangent to the velocity vector of the flow, so that

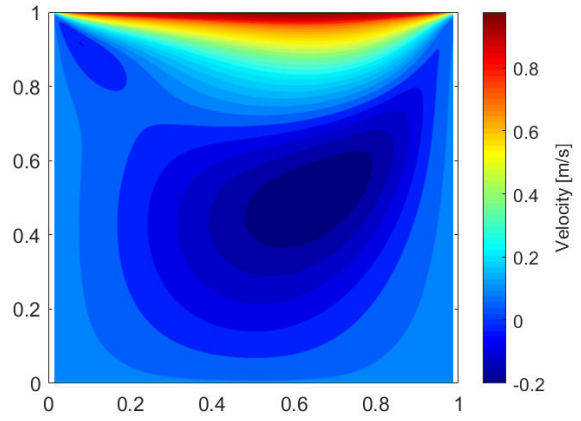
$$\frac{u}{v} = \frac{\mathrm{d}x}{\mathrm{d}y} \tag{5.2.24}$$

Combining Eq. 5.2.24 with Eqs. 5.2.22 and 5.2.23 leads to:

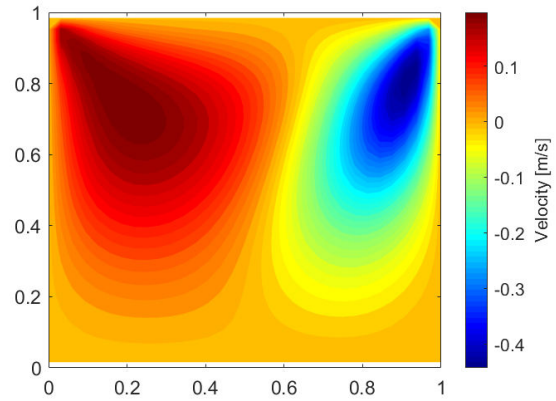
$$\frac{\partial \Psi}{\partial x} dx + \frac{\partial \Psi}{\partial y} dy = 0 \tag{5.2.25}$$

Then, the function is constant along a streamline and the stream function is completely defined except for an integration constant, which it is set to zero at the coordinates origin:

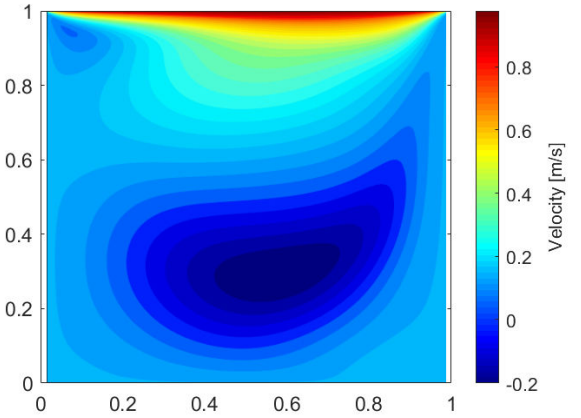
$$\Psi(x,y) = \int_0^y u \delta y - \int_0^x v \delta x - \int_0^x \left(\int_0^y \frac{\partial u}{\partial x} \delta y \right) \delta x \tag{5.2.26}$$



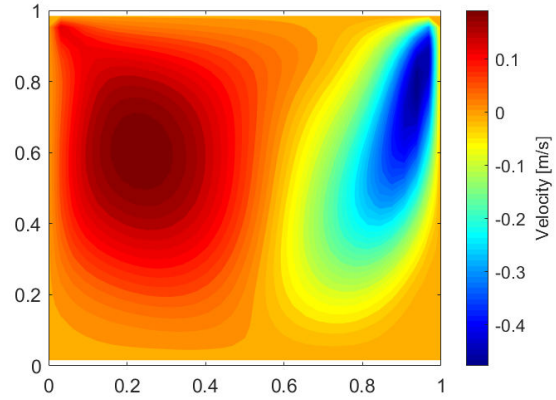
(a) u, $Re = 100$



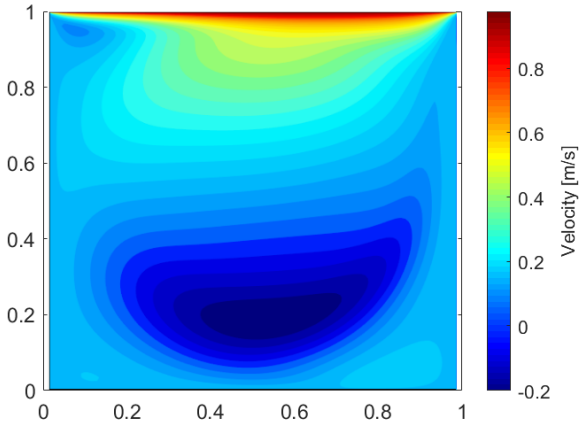
(b) v, $Re = 100$



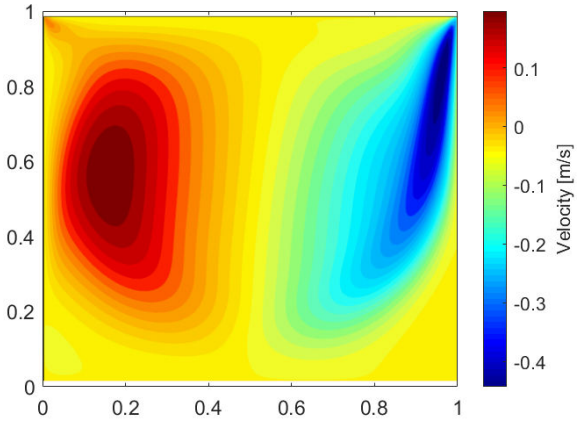
(c) u, $Re = 400$



(d) v, $Re = 400$

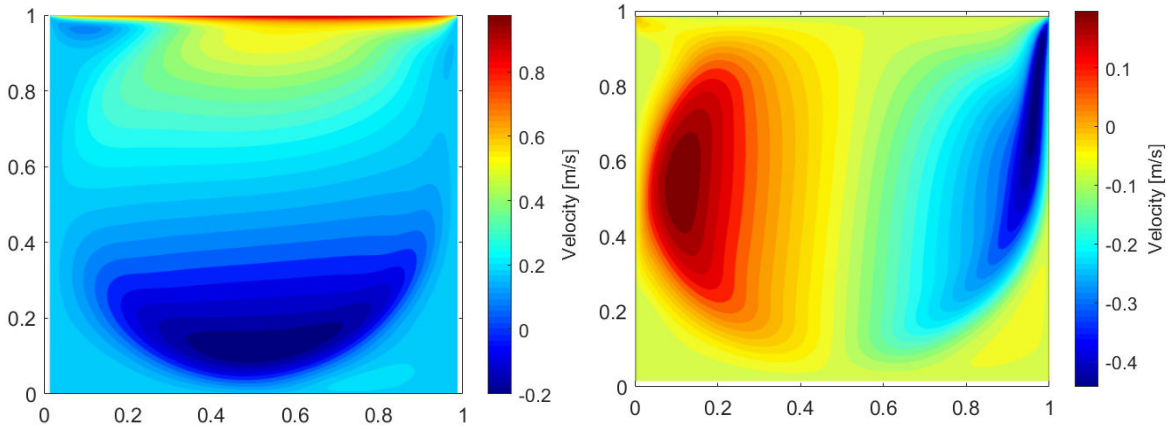


(e) u, $Re = 1000$



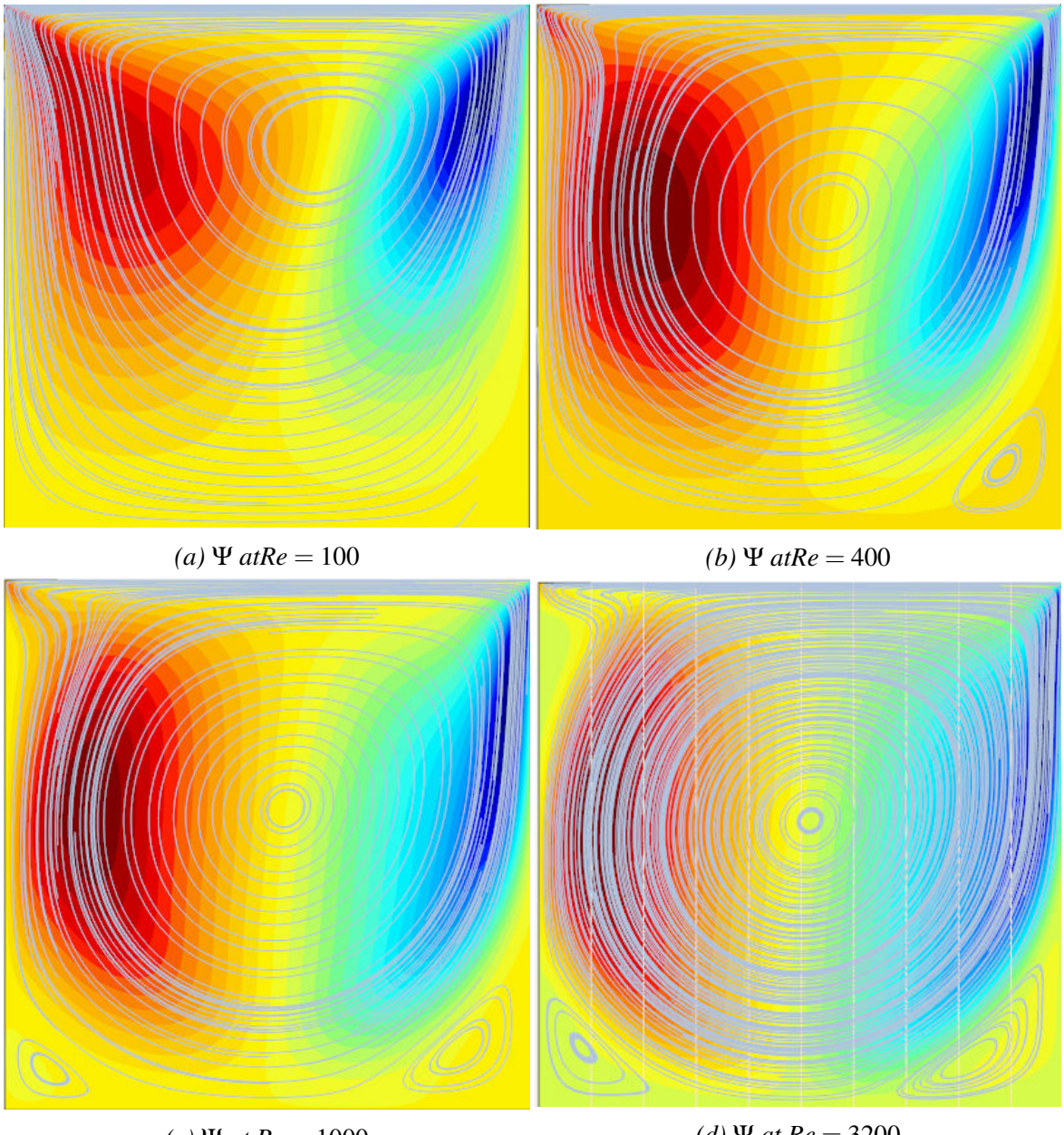
(f) v, $Re = 1000$

Figure 44: Velocity components u and v contour plots for different Re numbers



(a) u, Re = 3200 *(b) v, Re = 3200*

Figure 45: Velocity components u and v contour plots for different Re numbers



(a) Ψ atRe = 100 *(b) Ψ atRe = 400*

(c) Ψ at Re = 1000 *(d) Ψ at Re = 3200*

Figure 46: Stream function line plots for different Re numbers

By observing Figure 46 it is easy to see that the streamlines for lower Reynolds show a smoother flow, especially for $Re = 100$, where no strong eddies at the corners are visible. As Reynolds number increases, the centre of rotation moves towards the centre of the cavity and three eddies gain in strength. The first eddy makes its appearance at the lower-right corner of the cavity, followed by another at the lower-left corner while the last one appears at the upper-left corner.

5.2.6. Discussion

The results are overall satisfying, CDS and QUICK schemes reached convergence for $Re = 100, 400$ and 1000; their accuracy is comparable at lower Reynolds values. CDS is the simplest scheme of second-order accuracy and it offers a good compromise among accuracy and simplicity although its performance becomes worse when increasing the Reynolds number, if compared to QUICK scheme. UDS performs with the worst accuracy and it does not seem capable of handling this kind of problems.

In general, when Re increases, coarse grid could not be implemented in the method due to the raising of turbulent behaviour and mesh refinement becomes mandatory. For $Re > 3200$ it is not feasible to reach convergence because the computer time really increases, having the need to use a fine mesh of at least 200x200. Finally, it has been found that the relative error is significant near the walls; this could be solved refining the mesh, as said before, or using a non-uniform mesh with an higher density of CVS near the walls as shown in Figure 47.

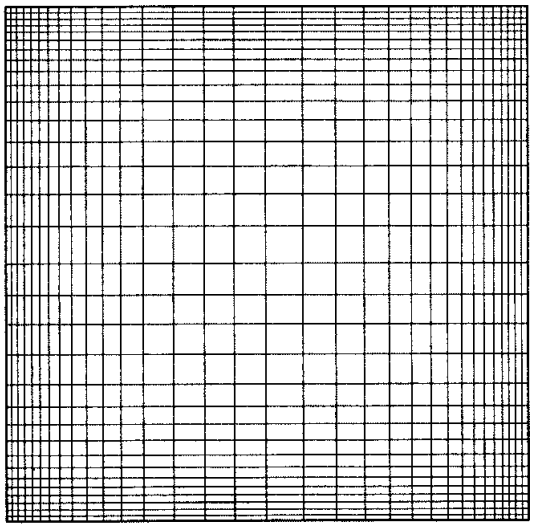


Figure 47: A non-uniform grid with 32x32 CVs, extracted from [12]

In this work, the implemented mesh is regular for simplicity purposes and for this reason the convergence of the results for high Re is limited, also due to the limits of the computer processor used.

5.3. The Differentially Heated Cavity (DHC) flow problem

Buoyancy-driven flow in a square cavity with vertical sides which are differentially heated is a suitable vehicle for testing and validating computer codes used for a wide variety of practical problems such as nuclear reactor insulation, ventilation of rooms, solar energy collection and crystal growth in liquids. [21].

5.3.1. Problem Definition

This problem consists of a fluid confined in a 2D square cavity with its lateral walls at different temperature and its upper and lower walls being adiabatic. The wall at the left side will remain at a higher temperature than the one at the right, thus leading the fluid to a clockwise rotation. Unlike in the Lid-Driven Cavity problem, where the upper wall moves at a certain speed, in this case any of the walls moves, so velocity is equally prescribed in all the cavity boundary being both of its component null. A sketch of the problem may be seen in the figure below.

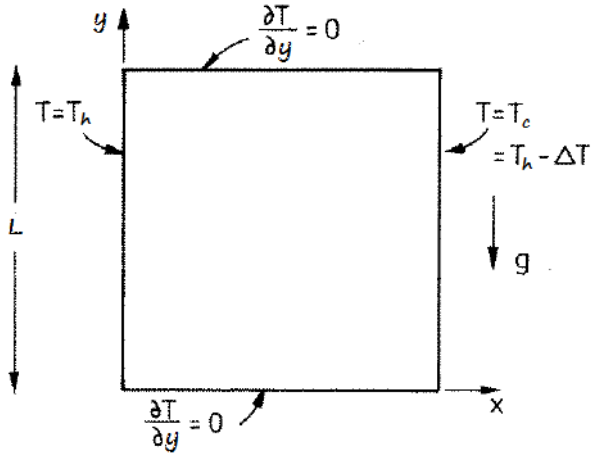


Figure 48: Sketch of the DHC problem

This is a case in which the source of the movement is the buoyancy term in the Navier-Stokes momentum equation. In such cases the Boussinesq approximation is applied. It basically consists in assuming that the variations in the fluid's density are neglected, thus leading to an incompressible flow, except in the volumetric forces, where this variations gain importance and are to be taken into account, since they are responsible for the fluid's movement.

5.3.2. Governing Equations

The equations involved in this problem are mass and momentum conservation (Navier-Stokes equations and continuity equation) along with an additional equation of energy con-

servation, not present in the Lid-Driven Cavity problem. The set of equations is:

$$\nabla \cdot \mathbf{v} = 0 \tag{5.3.1}$$

$$\rho \left(\frac{\partial \mathbf{v}}{\partial t} + (\mathbf{v} \cdot \nabla)\mathbf{v} \right) = -\nabla p + \mu \nabla^2 \mathbf{v} + \rho \mathbf{g} \tag{5.3.2}$$

$$\rho c_p \left(\frac{\partial T}{\partial t} + (\mathbf{v} \cdot \nabla)T \right) = k \nabla^2 T \tag{5.3.3}$$

Boussinesq approximation

The Boussinesq approximation is used in natural convection problems and it states that the density differences are small enough to be neglected, except where they have an influence on the buoyancy term (basically where they are multiplied by the acceleration of gravity g). This approximation replaces the density in the buoyancy term by an expression which depends on temperature. This dependency is achieved by introduction the thermal expansion coefficient β:

$$\rho = \rho_0 - \beta(T - T_0), \quad \beta > 0 \tag{5.3.4}$$

While the remaining desnities that appear in the equations will remain constant. Thus, equation 5.3.2 becomes:

$$\rho_0 \left(\frac{\partial \mathbf{v}}{\partial t} + (\mathbf{v} \cdot \nabla)\mathbf{v} \right) = -\nabla p + \mu \nabla^2 \mathbf{v} + \rho_0 \mathbf{g} - \beta(T - T_0)\mathbf{g} \tag{5.3.5}$$

Non-dimensional equations

It is convenient to convert the equations into non-dimensional expression in order to provide general and scaled solutions. The quantities which will be used to scale the variables of the problem are L,ρ ,c_p and k . The non-dimensional variables are then defined in the following expressions.

$$\widetilde{r} = \frac{r}{L} \tag{5.3.6}$$

$$\widetilde{\nu} = \frac{\nu}{\frac{\rho c_p L}{k}} \tag{5.3.7}$$

$$\widetilde{t} = \frac{t}{\frac{\rho c_p L^2}{k}} \tag{5.3.8}$$

$$\widetilde{p} = \frac{t}{\rho \left(\frac{c_p L}{k} \right)^2} \tag{5.3.9}$$

$$\widetilde{T} = \frac{T - T_c}{T_h - T_c} \tag{5.3.10}$$

Substituting the dimensional variables into equations 5.3.1, 5.3.2 and 5.3.3 the following non-dimensional equations can be obtained. The subscript "$\widetilde{\ }$" is omitted for simplicity pur-

poses. From now on, it is assumed that all the equations and variables are non-dimensional.

$$\nabla \cdot \mathbf{v} = 0 \tag{5.3.11}$$

$$\frac{\partial \mathbf{v}}{\partial t} + (\mathbf{v} \cdot \nabla)\mathbf{v} = -\nabla p + Pr\nabla^2 \mathbf{v} + Pr\, Ra\, T\, \hat{u}_g \tag{5.3.12}$$

$$\frac{\partial T}{\partial t} + (\mathbf{v} \cdot \nabla)T = k\nabla^2 T \tag{5.3.13}$$

Where $\hat{u}_g$ is a unit vector indicating the direction of the acceleration of gravity. Two non-dimensional numbers arise from this derivation: the Prandtl Pr number and the Rayleigh Ra number. It should be noted that the non-dimensional solution of the problem depends only on these two numbers, which for this problem are defined as:

$$Pr = \frac{c_p \mu}{k} \tag{5.3.14}$$

$$Ra = \frac{g\beta}{\mu \frac{k}{\rho c_p}} (T_h - T_c) L^3 \tag{5.3.15}$$

Generally speaking, the Prandtl number is a dimensionless parameter representing the ratio of diffusion of momentum to diffusion of heat [22]. Contrary to other dimensionless numbers such as the Reynolds number, Pr number does not contain any length scales in its definition and it depends only on the fluid and the fluid state. On the other hand, the Rayleigh number is associated with buoyancy-driven flow and it describes the behaviour of fluids when the mass density of the fluid is non-uniform [23].

5.3.3. Code development

Problem assumption

The assumptions are the same of the Lid-Driven Cavity flow problem, except for the fact that the density is not a constant properties

Mesh generation

Three meshes are created in a similar way of the previous problem. The main mesh is used for the scalar values of pressure and temperature while the other two staggered grids are created for the calculation of the velocity field.

Code equations

In this problem, the time integration has to be performed for two different equations: momentum and energy conservation. For the first equations, the same methodology used in the Lid-Driven Cavity problem is adopted while the Energy equation can be solved implementing an implicit or an explicit scheme.

Momentum Equation

It is necessary to define a new term containing the diffusive term and the flotation term. Such

term is so defined:

$$R(\mathbf{v}) = -(\mathbf{v}\cdot\nabla)v + Pr\nabla^2\mathbf{v} + Pr\,Ra\,T\,\hat{u}_g \tag{5.3.16}$$

It differs from the term 5.1.13 for the addition of the source term resulting for the Bousinessq approximation. The rest of the steps are analogous as described in section 5.1.2. The main difference is on the limitations on the time step. For this problem, the CFL condition becomes:

$$\Delta t \left(\frac{|u_i|}{\Delta x_i}\right)_{max} \leq C_{conv} \tag{5.3.17}$$

$$\Delta t \left(\frac{Pr}{\Delta x_i^2}\right)_{max} \leq C_{visc} \tag{5.3.18}$$

Where C_{conv} and C_{visc} are again 0.35 and 0.2 respectively.

Energy Equation

As it has been said in before, the energy equation may be solved using an implicit scheme as much as an explicit one. If an explicit scheme is to be used, a second order Adams-Bashforth approximation could be equally used. However, an implicit scheme is preferable, as it avoids numerical instabilities and it has been found the execution is not so higher. So, integrating equation 5.3.12 with respect to time:

$$\int_{t^n}^{t^{n+1}} \frac{\partial T}{\partial t} dt = T^{n+1} - T^n = \int_{t^n}^{t^{n+1}} (-(\mathbf{v}\cdot\nabla)T + \nabla^2 T)dt \approx [-(\mathbf{v}^{n+1}\cdot)T^{n+1} + \nabla^2 T^{n+1}]\Delta t$$

Rearranging the terms:

$$\frac{T^{n+1} - T^n}{\Delta t} = -(\mathbf{v}^{n+1}\cdot\nabla)T^{n+1} + \nabla^2 T^{n+1} \tag{5.3.19}$$

It is obvious that the velocity field is needed in order to obtain the temperature distribution in the whole domain. Therefore, momentum equation must be solved in first place. Once the velocity field is obtained, the energy equation can be solved.

Spatial-integration

Momentum equation

Momentum equation's discretization follows the same procedure explained in section 5.2.2.

Energy Equation

Integrating equation 5.3.19 over a certain control volume:

$$\int_{V_{cv}} \frac{T^{n+1} - T^n}{\Delta t} dV_{cv} = -\int_{V_{cv}} (\mathbf{v}^{n+1}\cdot\nabla)T^{n+1} dV_{cv} + \int_{V_{cv}} \nabla^2 T^{n+1} dV_{cv} \tag{5.3.20}$$

For the sake of clarity the different terms are developed separately.

Transient term

$$\int_{V_{cv}} \frac{T^{n+1}-T^n}{\Delta t} dV_{cv} = \frac{T_P^{n+1}-T_P^n}{\Delta t} V_{cv} \tag{5.3.21}$$

Convective term

$$\begin{gathered} \int_{V_{cv}} \nabla \cdot (\mathbf{v}^{n+1} T^{n+1}) dV_{cv} = \oint_{S_{cv}} \mathbf{v}^{n+1} T^{n+1} \cdot \hat{n}\, dS_{cv} \\ \approx F_e^{n+1} T_e^{n+1} + F_n^{n+1} T_n^{n+1} - F_w^{n+1} T_w^{n+1} - F_s n + 1 T_s^{n+1} \end{gathered} \tag{5.3.22}$$

Diffusive term

$$\begin{gathered} \int_{V_{cv}} \nabla \cdot (\nabla T^{n+1}) dV_{cv} = \oint_{S_{cv}} \nabla T^{n+1} \cdot \hat{n} dS_{cv} \approx \\ \approx \left(\frac{\partial T^{n+1}}{\partial x}\bigg|_e - \frac{\partial T^{n+1}}{\partial x}\bigg|_w \right) \Delta y + \left(\frac{\partial T^{n+1}}{\partial x}\bigg|_n - \frac{\partial T^{n+1}}{\partial x}\bigg|_s \right) \Delta x \approx \\ \approx \left(\frac{T_E^{n+1}-T_P^{n+1}}{\Delta x_{d_e}} - \frac{T_P^{n+1}-T_W^{n+1}}{\Delta x_{d_w}} \right) \Delta y + \left(\frac{T_N^{n+1}-T_P^{n+1}}{\Delta y_{d_n}} - \frac{T_P^{n+1}-T_S^{n+1}}{\Delta y_{d_s}} \right) \Delta x \end{gathered} \tag{5.3.23}$$

If the variables at the walls of the cell are computed using a CDS scheme, then a simple expression of the following form may be derived for each control volume:

$$a_P T_P^{n+1} + a_N T_N^{n+1} + a_S T_S^{n+1} + a_E T_E^{n+1} = b_P \tag{5.3.24}$$

Where the coefficients are:

$$a_P = \frac{\Delta x \Delta y}{\Delta t} + \frac{\Delta y}{\Delta x_{d_e}} + \frac{\Delta x}{\Delta y_{d_n}} - \frac{\Delta x}{\Delta y_{d_s}} + \frac{F_e^{n+1}}{2} + \frac{F_n^{n+1}}{2} - \frac{F_w^{n+1}}{2} - \frac{F_s^{n+1}}{2} \tag{5.3.25}$$

$$a_E = -\frac{\Delta y}{\Delta x_{d_e}} + \frac{F_e^{n+1}}{2} \tag{5.3.26}$$

$$a_W = -\frac{\Delta y}{\Delta x_{d_w}} - \frac{F_w^{n+1}}{2} \tag{5.3.27}$$

$$a_N = -\frac{\Delta x}{\Delta y_{d_n}} + \frac{F_n^{n+1}}{2} \tag{5.3.28}$$

$$a_S = -\frac{\Delta x}{\Delta y_{d_s}} - \frac{F_s^{n+1}}{2} \tag{5.3.29}$$

$$b_P = \frac{T_P^n \Delta x \Delta y}{\Delta t} \tag{5.3.30}$$

These coefficients are valid for every internal control volumes. Regarding the boundary conditions the coefficient must be slightly modified, as always. In the case of the upper or lower walls of the cavity, where the walls are adiabatic, the condition imposed is:

$$\frac{\partial T}{\partial y} = 0 \tag{5.3.31}$$

Regarding the volumes in contact with the left and right walls, where the temperature is prescribed, the coefficients are:

$$a'_P = 1, \quad a'_{i \neq P} = 0, \quad b'_P = T_c \quad if \ \ east \ \ side \tag{5.3.32}$$

$$a'_P = 1, \quad a'_{i \neq P} = 0, \quad b'_P = T_h \quad if \ \ west \ \ side \tag{5.3.33}$$

5.3.4. Verification

The verification is made using the benchmark solution provided by G. Davis [21]. These results are, in order of appearance:

- $|\Psi|_{mid}$ the stream function at the mid-point of the cavity;
- $|\Psi|_{max}$ the maximum absolute value of the stream function and its locatiomn;
- u_{max} the maximum horizontal velocity on the vertical mid-plane of the cavity together with its location;
- v_{max} the maximum vertical velocity on the horizontal mid-plane of the cavity together with its location;
- $\overline{Nu}$ the average Nusselt number throughout the cavity;
- $Nu_{1/2}$ the average Nusselt number on the vertical boundary at $x = 0$ together with its location;
- Nu_{max} the maximum value of the local Nusselt number on the boundary at $x = 0$ together with its location;
- Nu_{min} the minimum value of the local Nusselt number on the boundary at $x = 0$ together with its location;

The Nusselt number (Nu) is a dimensionless number and it represents the ratio of convective to conductive heat transfer at a boundary in a fluid. A Nusselt number of value one represents heat transfer by pure conduction, between 1 and 10 is characteristic of laminar flow while turbulent flow usually are defined by Nusselt numbers in the 100-1000 range.[24] Its calculation procedure is presented in the following steps [21]. The local heat flux in a horizontal

direction at any point in the cavity is:

$$q(x,y) = uT - \frac{\partial T}{\partial x} \tag{5.3.34}$$

If the heat flux is knwon for each point of the mesh the local heat transfer can be computed as:

$$h(x,y) = \frac{q(x,y)}{T_h - T_c} \tag{5.3.35}$$

and finally the local Nusselt number would be:

$$Nu(x,y) = \frac{h(x,y)L}{k} \tag{5.3.36}$$

Therefore, the *Nu* can be averaged depending on the desired extracted value. In Table 16 the results are compared with the benchmark solution.

	Ra							
	$\mathbf{10^3}$		$\mathbf{10^4}$		$\mathbf{10^5}$		$\mathbf{10^6}$	
$\mathbf{\lvert\psi\rvert_{mid}}$	1.174	1.177	5.071	5.110	9.111	9.157	16.320	16.784
$\mathbf{\lvert\psi\rvert_{mid}}$	-	-	-	-	9.612	9.747	16.750	17.004
x	-	-	-	-	0.285	0.718	0.151	0.855
y	-	-	-	-	0.601	0.402	0.547	0.392
$\mathbf{u_{max}}$	3.649	3.660	16.178	16.192	34.73	34.789	64.63	65.121
y	0.813	0.825	0.823	0.834	0.855	0.871	0.850	0.872
$\mathbf{v_{max}}$	3.697	3.701	19.617	19.631	68.59	68.711	219.36	221.018
x	0.178	0.182	0.119	0.127	0.066	0.081	0.0379	0.042
$\mathbf{\overline{Nu}}$	1.118	1.118	2.243	2.248	4.519	4.543	8.800	8.931
$\mathbf{Nu_{1/2}}$	1.118	1.118	2.243	2.248	4.519	4.543	8.799	8.931
$\mathbf{Nu_0}$	1.117	1.118	2.238	2.248	4.509	4.543	8.817	8.930
$\mathbf{Nu_{max}}$	1.505	1.509	3.528	3.545	7.717	7.802	17.925	18.387
y	0.092	0.088	0.143	0.151	0.081	0.072	0.0378	0.034
$\mathbf{Nu_{min}}$	0.692	0.687	0.586	0.581	0.729	0.722	0.989	0.972
y	1.000	0.994	1.000	0.993	1.000	0.995	1.000	0.994

Table 16: Comparison between the obtained results with a 100 x 100 mesh and the benchmark solution [21]

In general, the obtained results are similar to the ones given in [21]. However, there are some discrepancies. All the results are over estimated respect to the benchmark ones. Moreover, for greatest two values of *Ra* number, the location of the maximum absolute value of the stream function really differs from the benchmark value. This could be explained considering the symmetry of the flow: a small error on the value of the stream function

could change the position from one side to the other of the cavity. For this reason, the relative error can be reasonably computed with the corresponding symmetric value respect to the plane $y = x$. For the first two values of Rayleigh number, the maximum absolute value of the stream function are located at the centre of the cavity (this is the reason why no values are shown in the table). The relative errors are shown in Table 17.

	Ra			
-	$\mathbf{10^3}$	$\mathbf{10^4}$	$\mathbf{10^5}$	$\mathbf{10^6}$
$\lvert\psi\rvert_{\mathbf{mid}}$	0.175	0.123	0.401	1.551
$\lvert\psi\rvert_{\mathbf{max}}$	-	-	0.382	1.451
x	-	-	0.001	4.002
y	-	-	0.684	0.375
$\mathbf{u_{max}}$	0.033	0.032	0.121	0.714
y	0.881	0.871	0.603	1.204
$\mathbf{v_{max}}$	0.034	0.022	0.168	0.774
x	1.147	0.911	6.074	5.561
$\overline{\mathbf{Nu}}$	0.000	0.212	0.441	1.477
$\mathbf{Nu_{1/2}}$	0.000	0.212	0.441	1.482
$\mathbf{Nu_0}$	0.098	0.444	0.667	1.257
$\mathbf{Nu_{max}}$	0.145	0.327	1.099	2.347
y	7.701	1.421	7.502	7.502
$\mathbf{Nu_{min}}$	0.154	0.198	0.287	1.655
y	0.502	0.502	0.503	0.503

Table 17: Relative error (%) between the obtained results with a 100 x 100 mesh and the benchmark values [21]

5.3.5. Results and discussion

The results presented in this section correspond to air or a fluid with similar properties, since the Prandtl number has been set to 0.71. The most refined mesh is a grid of 100x100 nodes, identically to the one used for the Lid-Driven Cavity flow. Four cases are presented, corresponding to 4 different Rayleigh numbers $Ra = 10^3,\ 10^4,\ 10^5,\ 10^6$,

Contour plots

Temperature contour plots of Fig. 48 show the influence of the *Ra* on the temperature field: for low values of *Ra* the temperature field is stratified, showing almost a linear increment from the hot wall at the left to the cold one. When *Ra* increases, the effect of convection becomes more visible and forces the fluid to rotate clockwise: fluid in contact or near the left wall heats up and consequently decreases its density, causing its upward movement due to the buoyancy forces, while the fluid near the right wall is cooled and its density decreases,

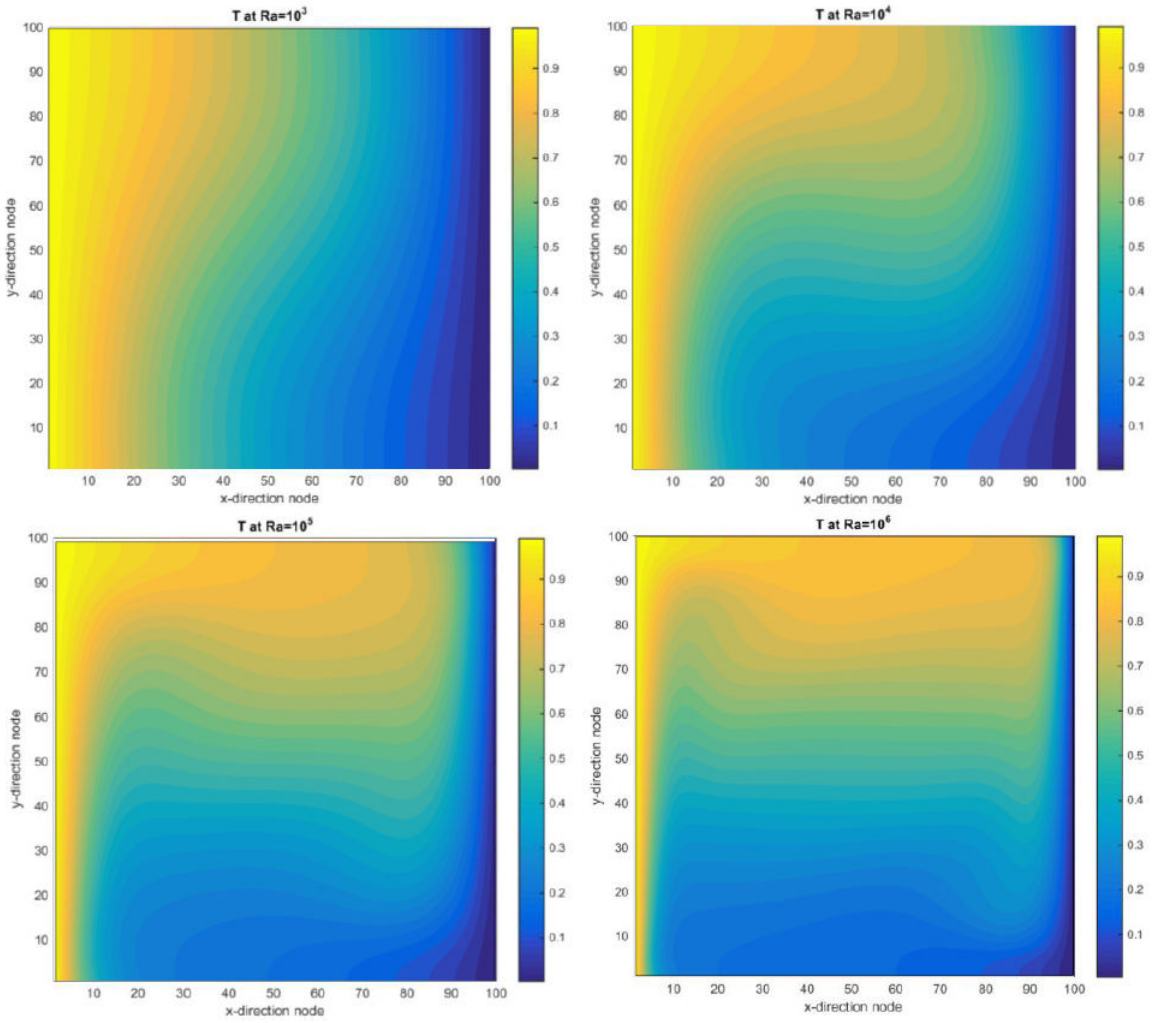


Figure 49: Temperature contour plots for different Ra numbers

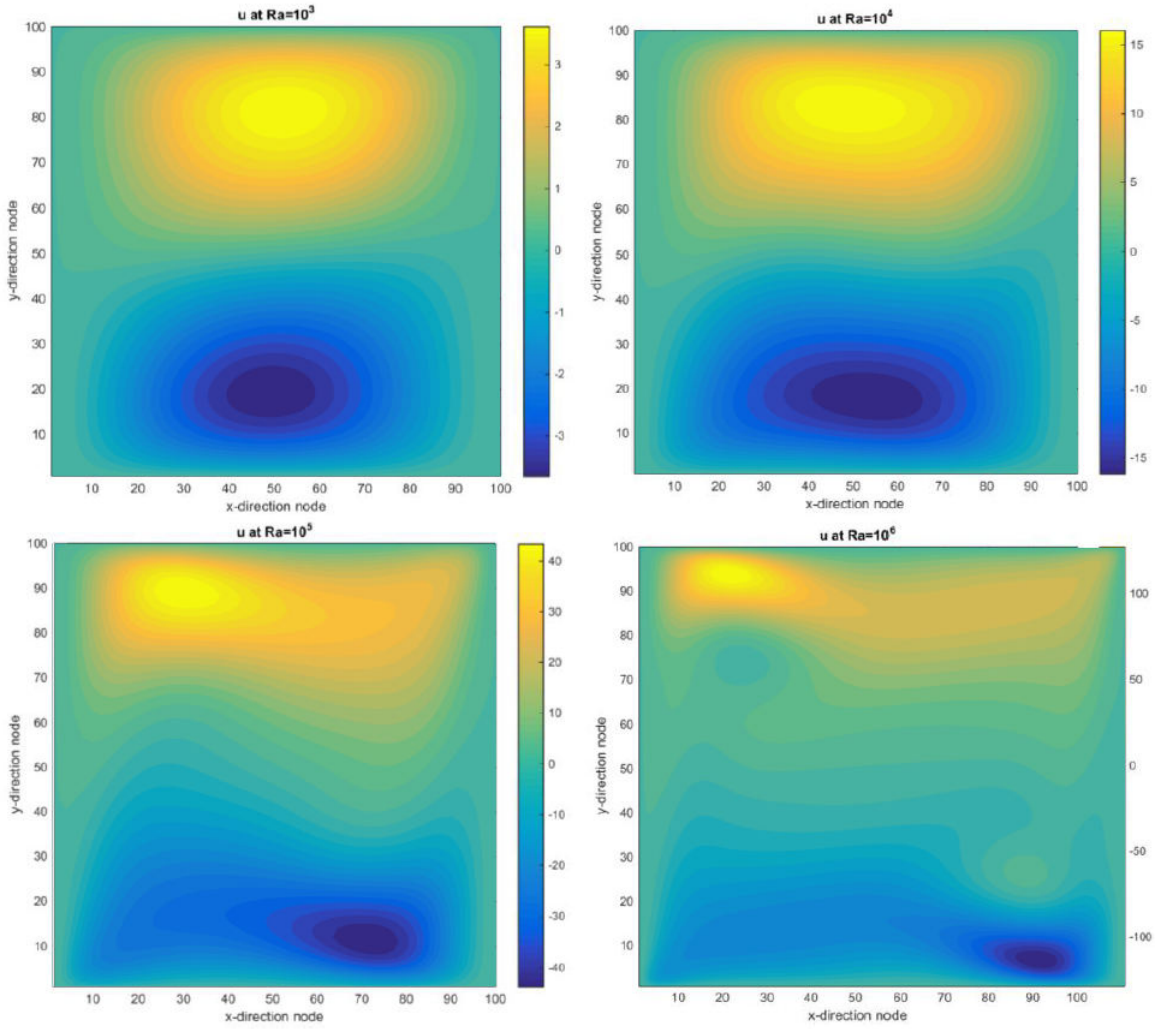


Figure 50: Horizontal velocity component u contour plots for different Ra numbers

causing its downward movement. . As the *Ra* increases more, the fluid's recirculation becomes stronger and the mixing of the fluid is more effective. At $Ra = 10^6$ temperature is transported mainly because of convection while the effect of conduction becomes negligible.

In figures 50 and 51 the contour plots for the velocity fields are shown.

It can be clearly seen that the horizontal component velocity field agrees with the clockwise motion of the flow. At lower *Ra* the maximum velocity magnitudes are found in the symmetry axis of the cavity. As the *Ra* increases, the velocity magnitude maximum increases too and it is found that the maximum values of velocity tend to move to the corners.

The tendency of the vertical component of the velocity field is very similar to the horizontal velocity field. Positive values (upwards motion) are located at the left wall while negative values (downwards motion) are located at the right wall of the cavity. The main effect of increase *Ra* is the stretching of the areas where velocity is higher, leading to high velocity gradients near the walls. It can be noted that at the centre of the cavity the velocities have the smallest values, similarly to horizontal velocity.

In Figure 52 the streamline field is presented.

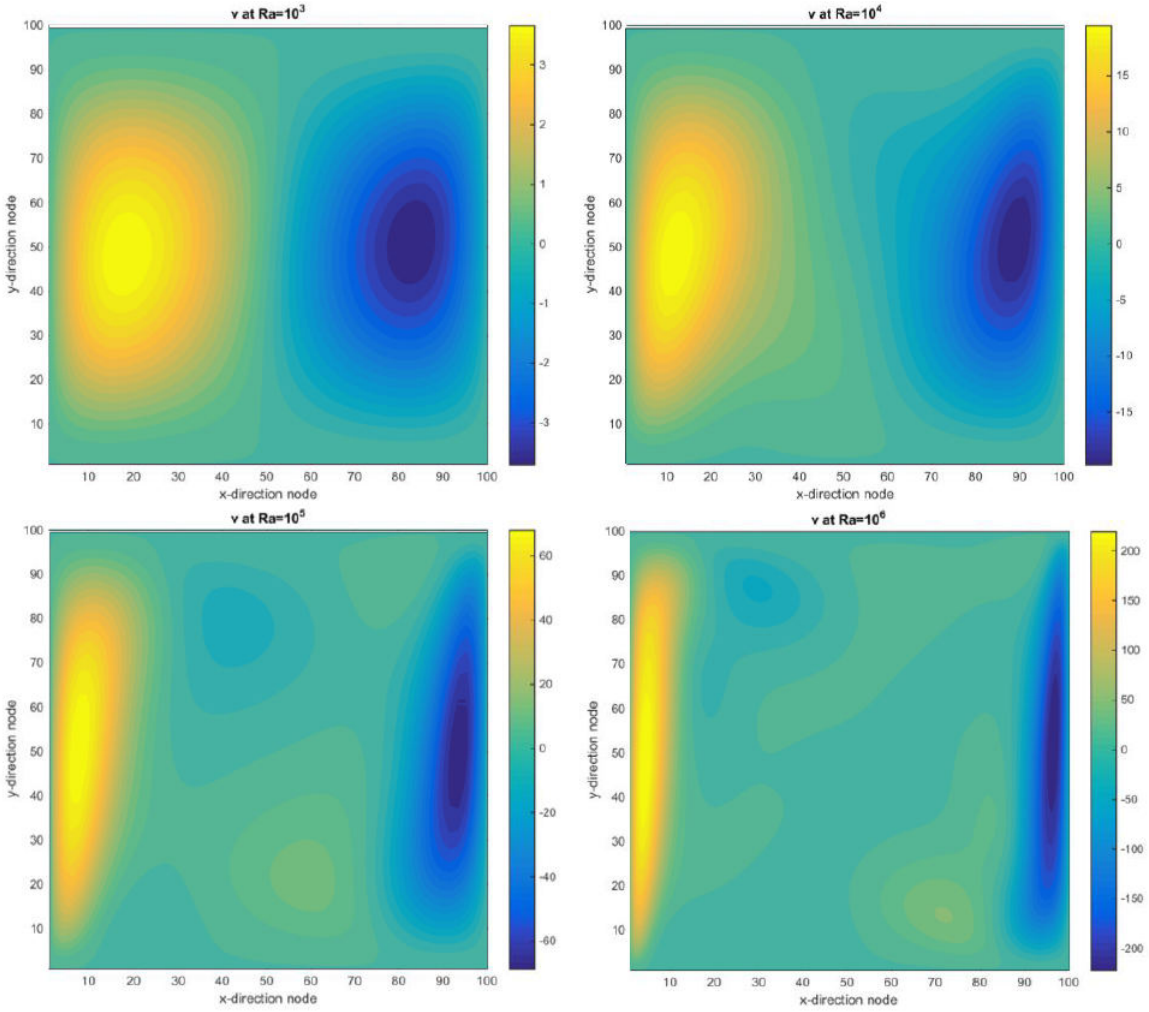
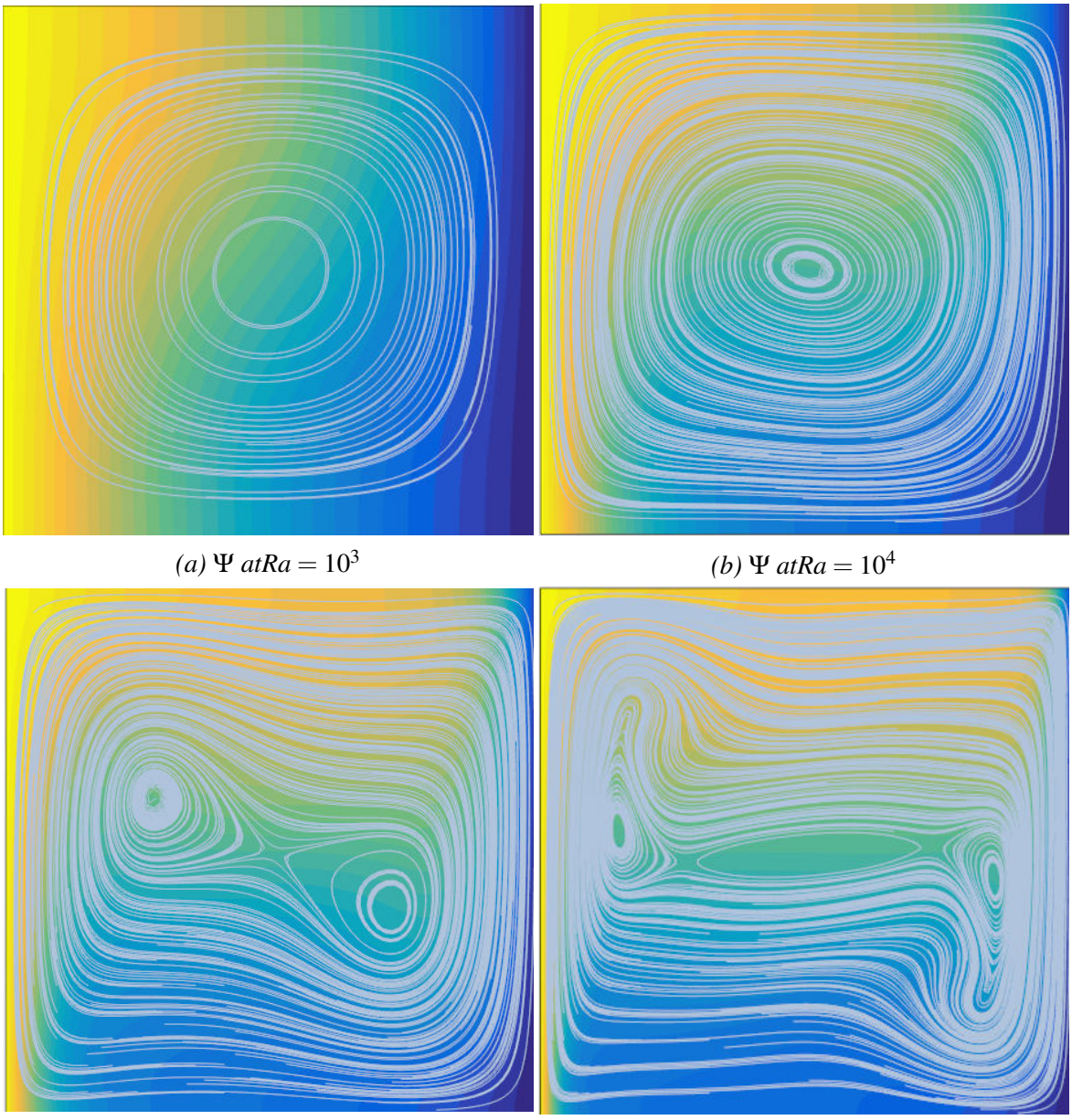


Figure 51: Vertical velocity component v plots for different Ra numbers



(a) Ψ $atRa = 10^3$

(b) Ψ $atRa = 10^4$

(c) Ψ $at\ Ra = 10^5$

(d) Ψ $at\ Ra = 10^6$

Figure 52: Streamline plots for different Ra numbers

Streamlines represent the clockwise motion of the fluid inside the cavity. For the first two values of *Ra* the fluid movement is very similar, though in the second case being it faster. For the third case, the main rotation vortex divides into two smaller ones. For the highest Rayleigh value the vortexes stretched to the corners forming a z-like shape. This can be explained with the fact that the fluid is heated up fast and gains such vertical velocities that it makes it difficult for the fluid to follow the rotation appropriately.

6. Numerical analysis of a flat-plate solar collector

6.1. Introduction

A solar collector is a heat exchanger that converts solar radiation into heat. It differs from a conventional heat exchanger for the fact that the latter usually perform a fluid-to-fluid exchange where radiation is an unimportant factor [25]. On the contrary, in solar collectors, energy is transferred from a distant source of radiant energy to a fluid. Flat-plate solar collectors are usually designed for applications that requires the fluid to go up to 100°C above the ambient temperature [25]. They use both beam and diffuse solar radiation and little maintenance is required. Flat-plate solar collector are usually found in solar water heating, buildings heating, air conditioning and industrial heat processes.

6.1.1. Description of flat-plate collectors

A typical liquid-heating flat-plate solar collector scheme is shown in Figure 53.

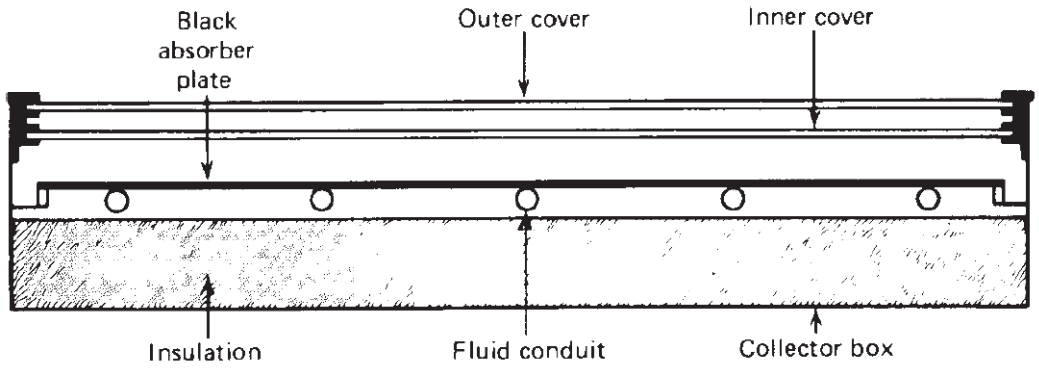


Figure 53: Cross section of a basic flat-plate solar collector [25].

The "black" absorber plate is the solar absorbing surface that transfers the absorbed energy to a fluid, the covers are envelopes applied to reduce convection and radiation losses to the ambient while the back insulation minimizes the conduction losses. Flat-plate solar collectors are almost always found in a fixed position, like a roof or a vertical wall, with an orientation optimized depending on the location and time of the year.

6.1.2. Basic energy balance equation

The performance of a solar collector can be described with an energy balance that shows how the incident solar energy is converted into useful energy gain, thermal losses, and optical losses. In particular, solar radiation absorbed by a collector per unit area of absorber S $[W/m^2]$ is equal to the difference between the incident solar radiation and the optical losses.

The thermal energy losses to the external ambient by conduction, convection, and infrared radiation is described with the product of an overall heat transfer coefficient U_L $\left[\frac{w}{m^2\,K}\right]$ times the difference between the mean absorber plate temperature T_{pm} $[K]$ and the ambient temperature T_a $[K]$. Therefore, at steady state, the useful energy of a collector of area A_c $[m^2]$ is the difference between the absorbed solar radiation and the thermal losses:

$$Q_u = A_c[S - U_L(T_{pm} - T_a)] \tag{6.1.1}$$

The problem of this apparently straightforward equation is that the mean absorber plate temperature is not easy to calculate or measure since it depends on many parameters such as the collector design, the solar radiation and the entering fluid condition. Therefore, Equation 6.1.1 have to be reformulated so that the useful energy can be calculated with the inlet fluid temperature and a parameter called the collector heat removal factor, which can be evaluated analytically or measured. The collector efficiency can be defined as the ratio of the useful heat produced in a time interval to the incident solar energy evaluated in the same time interval:

$$\eta = \frac{\int \dot{Q}_u dt}{A_c \int G_T dt} \tag{6.1.2}$$

If the conditions are assumed to be constant over in the time considered (usually one hour), the efficiency reduces to:

$$\eta = \frac{Q_u}{I_T A_C} \tag{6.1.3}$$

6.1.3. Collector overall heat loss coefficient

In order to simplify the study of the thermal losses in the solar collector, it is useful to introduce the concept of an overall loss coefficient. The thermal network of Figure 54 will be considered.

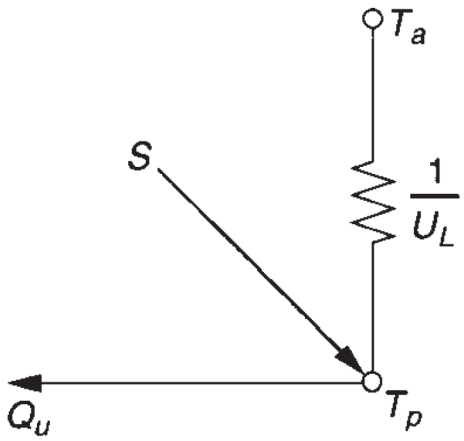


Figure 55: Equivalent thermal network for flat-plate solar collector [25]

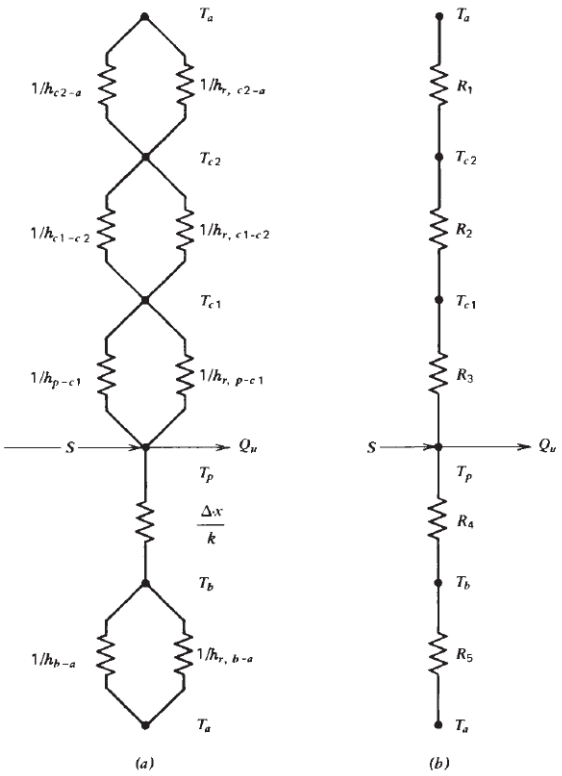


Figure 54: Thermal network for a two-cover flat-plate collector: (a) in terms of conduction, convection, and radiation resistances; (b) in terms of resistances between plates [25]

At some defined location on the plate where the temperature is T_P, solar energy of amount S is absorbed by the plate. The absorbed energy S divides into thermal losses through the top and bottom and to useful heat gain. The purpose of this section is to reduce the complex thermal network of Figure 54 to the simplest possible of Figure 55.

The energy loss through the top is due to the convection and radiation between the absorber plate and cover. The energy transfer between the plate at T_p and the first cover at T_{c1} is the

same as between two adjacent covers and is also equal to the energy losses to the ambient. Therefore, the loss through the top per-unit area can be evaluated through the heat transfer between the absorber plate to the first cover:

$$q_{loss,top} = h_{c,p-c1}(T_p - T_{c1}) + \frac{\sigma(T_p^4 - T_{c1}^4)}{\frac{1}{\varepsilon_p} + \frac{1}{\varepsilon_{c1}} - 1} \tag{6.1.4}$$

where $h_{c,p-c1}$ is the convection heat transfer coefficient, ε_p and ε_{c1} are the emissivities of the absorber plate and of the cover and σ is the Stefan–Boltzmann constant . Eq. 6.1.4 can be rearranged as:

$$q_{loss,top} = (h_{c,p-c1} + h_{r,p-c1})(T_p - T_{c1}) \tag{6.1.5}$$

where

$$h_{r,p-c1} = \frac{\sigma(T_p - T_{c1})(T_p^2 + T_{c1}^2)}{\frac{1}{\varepsilon_p} + \frac{1}{\varepsilon_{c1}} - 1} \tag{6.1.6}$$

The resistance R_3 can be then expressed as

$$R_3 = \frac{1}{h_{c,p-c1} + h_{r,p-c1}} \tag{6.1.7}$$

A similar expression can be written for R_2, the resistance between the covers. Although this theoretical analysis was performed for a collector with two covers, in reality most collector have only one cover. The resistance from the top cover to the surroundings has the same form as Equation 6.1.7. Even though the radiation resistance from the top cover exchange heat with the sky at T_s, this resistance is referenced to the ambient temperature for simplicity , so that the radiation heat transfer coefficient can be written as

$$h_{r,c2-a} = \sigma\varepsilon_c \frac{T_{c2}^4 - T_a^4}{T_{c2}^4 - T_a^4} \tag{6.1.8}$$

The resistance to the surroundings R_1 is then given by

$$R_1 = \frac{1}{h_{c,c2-a} + h_{r,c2-a}} \tag{6.1.9}$$

For this two-cover system, the top loss coefficient from the collector plate to the ambient is

$$U_t = \frac{1}{R_1 + R_2 + R_3} \tag{6.1.10}$$

The top loss coefficient has to be calculated with an iterative process, using Equations 6.1.4 through 6.1.35. The cover temperature has to be guessed at first, from which the convective and radiative heat transfer coefficients between the plate and cover can be calculate. With these estimates, Equation 6.1.35 can be solved for the top loss coefficient. The heat

losses from the collector top are evaluated with the top loss coefficient times the overall temperature difference, and since the energy exchange between plates corresponds also to the the overall heat loss, a new set of cover temperatures can be calculated with an energy balance. Beginning at the absorber plate, a new temperature is calculated for the first cover. This new first cover temperature is used then to find the next cover temperature, and so on. For a general pair of tow plates i and j, the new temperature of plate or cover j can be expressed in terms of the temperature of plate or cover i as

$$T_j = T_j - \frac{U_t(T_p - T_a)}{h_{c,i-j} - h_{r,i-j}} \tag{6.1.11}$$

The iterations continue until the cover temperatures do not change for a set tolearance.

The energy loss through the bottom of the collector is again modelled with two series of resistors, R_4 and R_5, in Figure 55, where R_4 represents the heat transfer resistance to the insulation and R_5 represents the convection and radiation resistance to the ambient. The magnitudes of R_4 and R_5 are such that it is usually possible to assume R_5 is zero and all resistance to heat flow is due to the insulation [25]. The back loss coefficient U_b can be evaluated as:

$$U_b = \frac{1}{R_4} = \frac{k}{L} \tag{6.1.12}$$

where k and L are the insulation thermal conductivity and thickness, respectively. For most collectors the evaluation of edge losses is complicated. However, in a well-designed system, the edge loss should be small so that it is not necessary to predict it with great accuracy [25]. The edge losses are assumed to occur around the perimeter of the collector system. The losses through the edge should be referenced to the collector area. If the edge loss coefficient–area product is $(U/A)_{edge}$, then the edge loss coefficient, based on the collector area A_c, is

$$U_e = \frac{(UA)_{edge}}{A_c} \tag{6.1.13}$$

If it is assumed that all heat losses occurs to the same sink temperature T_a, the collector overall loss coefficient U_L contains the top, bottom, and edge loss coefficients:

$$U_L = U_t + U_b + U_e \tag{6.1.14}$$

6.1.4. Temperature distribution between tubes and the collector efficiency factor

The temperature distribution between two tubes can be derived with the temporary assumption that the temperature gradient in the flow direction is negligible. The sheet-tube configuration is shown in Figure 56. The distance between the tubes is W, the tube diameter is D, and the sheet is thin with a thickness δ. The temperature gradient through the sheet is

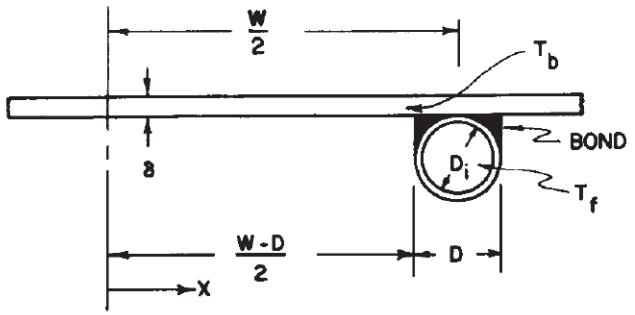


Figure 56: Sheet and tube dimensions [25]

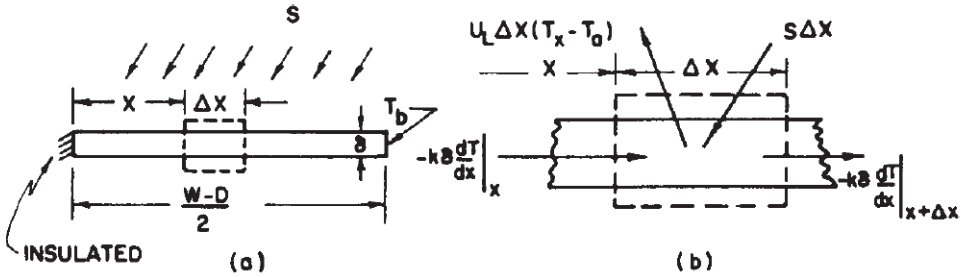


Figure 57: Energy balance ona fin element [25]

considered negligible, being the material a good conductor. The sheet above the bond is at a base temperature T_b. The region between the center-line separating the tubes and the tube base is treated as a classical fin problem [25]. The fin, shown in Figure 57(a), is of length $(W-D)/2$. An elemental region of width Δx and unit length in the flow direction is shown in Figure 57(b). An energy balance on this element yields

$$S\Delta x - U_L\Delta x(T-T_a) + \left(-k\delta\frac{\mathrm{d}T}{\mathrm{d}x}\right)\Big|_x - \left(-k\delta\frac{\mathrm{d}T}{\mathrm{d}x}\right)\Big|_{x+\Delta x} = 0 \tag{6.1.15}$$

where S is the adsorbed solar energy previously defined. Dividing through by Δx and finding the limit as Δx approaches zero yield

$$\frac{\mathrm{d}^2T}{\mathrm{d}x^2} = \frac{U_L}{k\delta}\left(T - T_a - \frac{S}{U_L}\right) \tag{6.1.16}$$

The two boundary conditions are symmetry at the center-line and the known base temperature T_b:

$$\frac{\mathrm{d}T}{\mathrm{d}x}|_{x=0} = 0, \quad T|_{x=(W-D)/2} = T_b \tag{6.1.17}$$

The detailed solution of the equation can be found in [25]. The energy conducted to the region of the tube per unit of length in the flow direction can now be found by evaluating Fourier's law at the fin base:

$$q'_{fin} = (W-D)[S - U_L(T_b - T_a)]\frac{tanh\ m(W-D)/2}{m(W-D)/2} \tag{6.1.18}$$

where, for convenience, m is defined as:

$$m = \sqrt{\frac{U_L}{k\delta}} \tag{6.1.19}$$

It is convenient to use the concept of a fin efficiency to rewrite Equation 6.1.18 as

$$q'_{fin} = (W-D)F[S-U_L(T_b-T_a)] \tag{6.1.20}$$

where

$$F = \frac{tanh[m(W-D)/2]}{m(W-D)/2} \tag{6.1.21}$$

The useful gain of the collector also includes the energy collected above the tube region. The energy gain for this region is

$$q'_{tube} = D[S-U_L(T_b-T_a)] \tag{6.1.22}$$

and the useful gain for the tube and fin per unit of length in the flow direction is the sum of Equations 6.1.20 and 6.1.22:

$$q'_u = [(W-D)F+D][S-U_L(T_b-T_a)] \tag{6.1.23}$$

The total useful gain from Equation 6.1.23 is transferred to the fluid through the heat flow resistance between the bond and the tube-to-fluid resistance. The useful gain can be expressed in terms of the two resistances as

$$q'_u = \frac{T_b - T_f}{\frac{1}{h_{fi}\pi D_i} + \frac{1}{C_b}} \tag{6.1.24}$$

where D_i is the inside tube diameter and h_{fi} is the heat transfer coefficient between the fluid and the tube wall. The bond conductance C_b can be estimated from knowledge of the bond thermal conductivity k_b, the average bond thickness γ and the bond width b [25]. On a per-unit-length basis,

$$C_b = \frac{k_b b}{\gamma} \tag{6.1.25}$$

The base temperature T_b has to be eliminated from the equations and obtain an expression for the useful gain in terms of dimensions, physical parameters and the local fluid temperature. Solving Equation 6.1.26 for T_b, substituting it into Equation 6.1.23, and solving the result for the useful gain yields:

$$q'_u = WF'[S-U_L(T_f-T_a)] \tag{6.1.26}$$

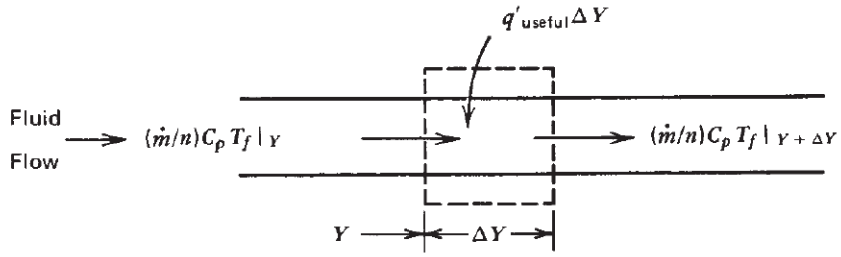


Figure 58: Energy balance on a fluid element [25]

where the collector efficiency factor F' is given as:

$$F' = \frac{1/U_L}{W\left[\frac{1}{U_L[D+(W-D)F]} + \frac{1}{C_b} + \frac{1}{\pi D_i h_{fi}}\right]} \tag{6.1.27}$$

At a particular location, F' is the ratio of the actual useful energy gain to the useful gain that would result if the absorber had been at the local fluid temperature T_f. Another interpretation for the parameter F' becomes from the denominator of Equation 6.1.27, which is the heat transfer resistance from the fluid to the ambient air. This resistance will be given the symbol $1/U_o$. The numerator is the heat transfer resistance from the absorber plate to the ambient air so that F' is the ratio of these two heat transfer coefficients:

$$F' = \frac{U_o}{U_L} \tag{6.1.28}$$

The collector efficiency factor is essentially a constant for any collector design and fluid flow rate even though the factor F' is a function of U_l and h_{fi}, that have some sort of temperature dependence but it can be assumed negligible. [25].

6.1.5. Temperature distribution in flow direction

The useful gain per unit flow length as calculated from Equation 6.1.26 is the heat transferred to the fluid. The fluid enters the collector at temperature T_{fi} and increases in temperature until it exits at temperature T_{fo}.

Referring to Figure 58 an energy balance on the fluid flowing through a single tube of length Δy is performed:

$$\left(\frac{\dot{m}}{n}\right) C_p T_f|_y - \left(\frac{\dot{m}}{n}\right) C_p T_f|_{y+\Delta y} + \Delta y q'_u = 0 \tag{6.1.29}$$

where $\dot{m}$ is the total collector flow rate and n is the number of parallel tubes. Dividing by Δy and finding the limit as Δy approaches zero and substituting in Equation 6.1.26 for q'_u yields:

$$\dot{m}C_p \frac{\mathrm{d}T_f}{\mathrm{dy}} - nWF'[S - U_L(T_f - T_a)] = 0 \tag{6.1.30}$$

if F' and U_L are assumed independent of position, then the solution of the fluid temperature at any position y is:

$$\frac{T_f - T_a - S/U_L}{T_{fi} - T_a - S/U_L} = exp\left(-\frac{U_L n W F' y}{\dot{m} C_p}\right) \tag{6.1.31}$$

finally, if the collector has length L in the flow direction and the outlet fluid temperature is T_{fo} at $y = L$, the equation can be rearranged as:

$$\frac{T_{fo} - T_a - S/U_L}{T_{fi} - T_a - S/U_L} = exp\left(-\frac{U_L A_c F'}{\dot{m} C_p}\right) \tag{6.1.32}$$

where the quantity (nWL) is the collector area A_c.

6.1.6. Collector heat removal and flow factor

It is useful to define a parameter that establishes a relation between the actual useful energy gain of a collector to the useful gain if the whole collector surface were at the fluid inlet temperature [25]. This parameter is called the collector heat removal factor F_R. In mathematical form is defined as:

$$F_R = \frac{\dot{m} C_p (T_{fo} - T fi}{A_c [S - U_L (T_{fi} - T_a]} \tag{6.1.33}$$

with further passages [25] the equation can be written as:

$$F_R = \frac{\dot{m} C_p}{A_c U_L}\left[1 - exp\left(-\frac{A_c U_L F'}{\dot{m} C_p}\right)\right] \tag{6.1.34}$$

F_R defines the ratio of the actual heat transfer to the maximum possible heat transfer. The maximum possible useful energy gain in a solar collector occurs when the whole collector is at the inlet fluid temperature because heat losses to the surroundings are at a minimum, so that $F_R$times this maximum possible useful energy gain is equal to the actual useful energy gain Q_u:

$$Q_u = A_c F_R [S - U_L (T_i - T_a)] \tag{6.1.35}$$

With Equation 6.1.35, the useful energy gain is finally calculated as a function of the inlet fluid temperature, that was the scope of all the derivation. The inlet fluid temperature is usually known or it is easy to measure. The effect of the parameter F_R is to reduce the useful energy gain from what it would have been if the whole collector absorber plate had been at the inlet fluid temperature.

6.2. Analysis of a flat-plate solar collector with a one-dimensional model

The performance of a typical flat-plate solar collector can be simply analyzed with a one-dimensional model, derived thanks to all the considerations and equations of Section 6.1.

Latitude	Φ	41.39º
Day (*June* 15^{th})	n	167
Surface inclination	β	15.30º
Surface orientation	γ	0 (south)

Table 19: Input data for the solar incident calculation

The flat-plate collector is supposed to be located in the city of Barcelona (Spain). Its performances will be analysed during a typical summer day and with a fixed orientation towards the sun, the sky will be assumed clear.

6.2.1. Calculation of the instantaneous incident solar radiation

The objective of this section is to obtain the total incident solar radiation I_T for the desired day, necessary for the calculation of the parameter S previously mentioned. This data can be simply measured or retrieved from a weather data station and then elaborate though a radiation model to calculate the instantaneous incident solar radiation for the desired day and solar collector inclination. The Weather Data file is provided in a *.stat* extension where the average monthly global horizontal radiation is provided. The data is reported in Table 18.

January	6.80
February	9.65
March	13.88
April	18.54
May	22.25
June	24.03
July	23.37
August	20.42
September	16.46
October	11.40
November	7.73
December	6.04

Table 18: Monthly solar radiation average values for Barcelona [MJ/m^2day]

The input data for the calculation (and their description) are listed in Table 19

The day can be chose quite arbitrarily, however, June is the month with the highest average radiation while the day it was chosen for simplicity purposes (no interpolation between the data is needed). Moreover, the inclination of the collector is calculated in order to maximize the impacting solar radiation and the collector is obviously oriented to south to maximize the solar exposure time.

The clearness index is defined as:

$$K_t = \frac{H}{H_0} \tag{6.2.1}$$

H_0 is the extra-terrestrial radiation, evaluated with the following equation:

$$H_0 = \frac{86400 G_{sc}}{\pi}\left[1+0.033cos\left(\frac{360n}{365}\right)\right]\left(cos\Phi cos\delta sin\omega_s + \frac{\pi\omega_s}{180}sin\Phi sin\delta\right) \tag{6.2.2}$$

where:

1. $G_{sc} = 1367\ \frac{W}{m^2}$ is the solar constant;
2. δ is the declination of the sun. The declination is the angular position of the sun at solar noon with respect to the plane of the equator; it can be calculated with:

$$\delta = 23.45sin\left(360\frac{284+n}{365}\right) \tag{6.2.3}$$

3. ω_s is the sunset hour angle, defined by the relation

$$cos\ \omega_s = -\frac{sin\ \Phi\ sin\ \delta}{cos\ \Phi\ cos\ \delta} = -tan\ \Phi\ tan\ \delta \tag{6.2.4}$$

After obtaining the clearness index, the rateo of the diffuse radiation over the total can be evaluated as follows [25]:

$$\frac{H_d}{H} = \begin{cases} 0.9 & \text{for } K_t \leq 0.17 \\ 1.188 - 2.272K_t + 9.473K_t^2 - 21.865K_t^3 + 14.648K_t^4 & \text{for } 0.17 < K_t \leq 0.75 \\ -0.54K_t + 0.632 & \text{for } 0.75 < K_t \leq 0.80 \\ 0.2 & \text{for } K_t > 0.80 \end{cases} \tag{6.2.5}$$

Now the diffuse radiation H_d can be computed. The interest though, in analysing a solar collector, is in calculate the instantaneous radiation I and its components "beam", "diffuse" and "reflected", denoted by the subscripts b, d and ρ. From statistical studies is possible to obtain correlations which relate H and I [25]. Two coefficients are defined:

$$r_t = \frac{I}{H} \tag{6.2.6}$$

$$r_d = \frac{I_d}{H_d} \tag{6.2.7}$$

where:

$$r_t = \frac{\pi}{24}(a + b\ cos\ \omega)\frac{cos\ \omega - cos\ \omega_s}{sin\ \omega_s - \frac{\pi\omega_s}{180}\ cos\ \omega_s} \tag{6.2.8}$$

$$r_d = \frac{\pi}{24} \frac{cos\ \omega - cos\ \omega_s}{sin\ \omega_s - \frac{\pi \omega_s}{180} cos\ \omega_s} \tag{6.2.9}$$

$$a = 0.409 + 0.5016\ sin\ (\omega_s - 60) \tag{6.2.10}$$

$$b = 0.6609 - 0.4767\ sin\ (\omega_s - 60) \tag{6.2.11}$$

Now, the fact that the radiation is impacting on a tilted t surface has to be taken into account. The total instantaneous solar radiation can be calculated as:

$$I_t = I_{b,t} + I_{d,t} + I_{\rho,t} \tag{6.2.12}$$

The beam radiation is evalueated as:

$$I_{b,t} = I_b R_b \tag{6.2.13}$$

where

$$R_b = \frac{cos\ \theta}{cos\ \theta_z} \tag{6.2.14}$$

θ and θ_Z are respectively the incident angle of radiation for the tilted surface and for a horizontal surface, they are calculated with geometric considerations that can be found in [25]. Adopting an isotropic model, the diffuse radiation is:

$$I_{d,t} = I_d \left(\frac{1 + cos\ \beta}{2} \right) \tag{6.2.15}$$

and the reflected radiation is:

$$I_{\rho,t} = I\rho \left(\frac{1 - cos\ \beta}{2} \right) \tag{6.2.16}$$

where ρ is the albedo of the surrounding surfaces. With this simple radiation model, the instantaneous radiation for a tilted surface, in a chosen position and in a chosen day of the year can be evaluated. The results are shown in Figure 59.

6.2.2. Evaluation of the total heat loss coefficient

In order to calculate the total heat loss coefficient of a solar collector, an iterative procedure has to be performed as explained in section 6.1.3. The algorithm is explained below:

1. Assume a cover temperature T_c^*;
2. Evaluate the radiation heat transfer coefficients $h_{r,p-c}$ and $h_{r,c-a}$;
3. Evaluate the plate-cover convective heat transfer coefficient $h_{c,p-c} = Nu\frac{k}{L}$;
4. Evaluate the top loss coefficient U_t;

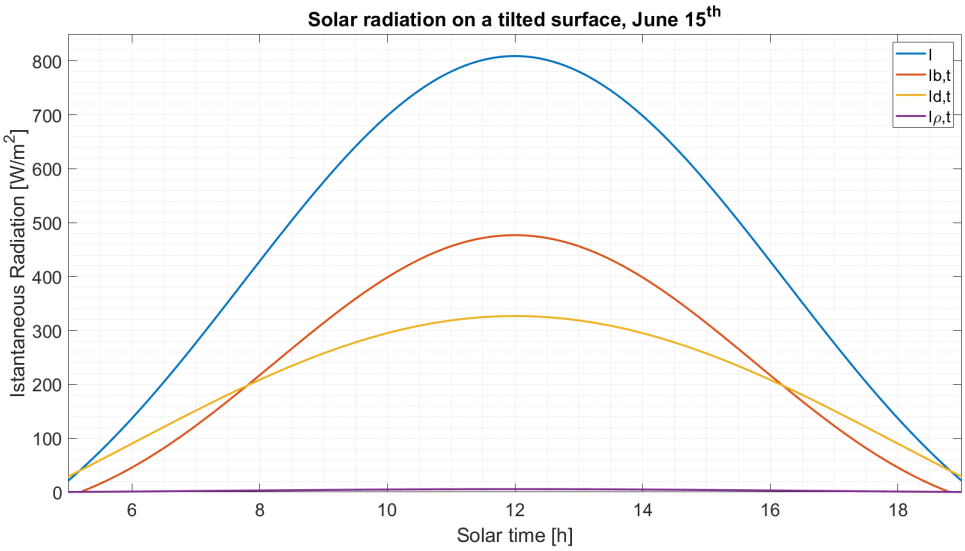


Figure 59: Instantaneous solar radiation for the solar collector, in Barcelona on June 15^{th}

5. Re-evaluate the cover temperature imposing an energy balance $q_{c-p} = q_{loss}$, leading to:

$$T_c = T_p - \frac{U_t(T_p - T_a)}{h_{c,p-c} + h_{r,p-c}} \tag{6.2.17}$$

6. if $T_c \neq T_c^* \rightarrow T_c^* = T_c$ and go to 2

7. Evaluate the convection and radiation fluxes as well as the total losses by the cover.

At point 3 there is a crucial evaluation for the correct analysis of the collector's performances. The Nu can be evaluated from a correlation or calculated with the numerical methods used in this work. In particular, the Nu number can be calculated with the methods explained in section 5.3, treating the tri-dimensional space between the absorber and the plate as a rectangular 2D cavity. Both the correlation and the numerical method will be implemented and their differences discussed. In the pure one-dimensional analysis performed in section 6.2.3, the Nu is calculated thanks to a correlation [25]:

$$Nu = 1 + 1.44\left[1 - \frac{1708(sin\,(1.8\beta))^1.6}{Ra\,cos\,\beta}\right]\left[1 - \frac{1708}{Ra\,cos\,\beta}\right] + \left[\left(\frac{Ra\,cos\,\beta}{5830}\right)^{1/3} - 1\right] \tag{6.2.18}$$

.

6.2.3. Results and discussion

A typical flat-plate collector is simulated under test conditions in order to determine the usual test parameters. The collector has the following characteristics:

Number of glass covers	1
Back-insulation thickness	0.070 m
Back-insulation thermal conductivity	0.0245 W/mK
Edge loss coefficient	0.098 W/m^2K
Plate-to-cover spacing	0.028 m
Circulating fluid	Water
Flow (based on gross area)	0.02 $kg/s\,m^2$
Number of tubes in parallel	8
Distance between tubes	0.115 m
Tube diameter (thin walled)	0.015m
Collector gross length	2.5m
Collector gross width	1.0 m
Bond conductance between tubes and absorber	∞
Absorber plate thickness	0.005 m
Absorber plate material	Copper
Absorber plate thermal conductivity	385 $W/m^2\,K$
Plate emittance curve [25]	$-0.00443 + 0.0003451 \cdot T + (2.6186 \cdot 10^{-7})T^2$
Cover emissivity	0.84
Cover $\tau\alpha$ product	0.88

Table 20: Input data for flat-plate solar collector simulation

The test procedure requires a minimum total solar radiation of 800 W/m^2 on the collector aperture with less than 20% diffuse solar radiation, a mass flow rate of 0.02 kg/sm^2, and a wind speed between 2 and 4 m/s. The collector is operated in Barcelona, Spain (latitude 41.4^o , longitude 2.2^o), on June 15th at a slope of 15.3^o (the optimal angle has been found with the radiation model) on a clear day, an ambient temperature of 20^{o}C in a climate with a relative humidity of 77%. The inlet temperature varies from 25 to 130^{o}C (at a pressure of 3 bar to avoid boiling). The wind speed over the collector and its mounting is 2.6 m/s. The ground reflectance is estimated to be 0.4. The simulation generates a plot of collector efficiency as a function of $(T_{in} - T_a)/I_T$ and determines values for the linear parameters given by Equation 6.17.3 and values for the second-order parameters given by Equation 6.17.7.

Solution

The solution algorithm is presented below:

1. Assume mean absorber plate temperature T_p^* and mean water temperature T_w^*;
2. Determine U_l with the algorithm of Section 6.2.2;
3. Solve for the temperature distribution along the tubes and determine the outlet water temperature T_{fo};

4. Determine the useful heat transfer to the fluid $Q_u = \dot{m}c_p(T_{fo} - T_{fi})$
5. Determine the new mean absorber plate T_p from an energy balance $T_p = \frac{(\tau\alpha)I_t - Q_u/A_c}{U_L} + T_a$
6. if $T_p \neq T_p^*$ go to step 1.

As with all systems of nonlinear equations, reasonable initial guesses are required. For this problem for all inlet fluid temperatures initial guesses for the mean plate temperature, mean fluid temperature and mean cover temperature are needed. In this simulated test all measurements are assumed to occur very close to noon, so that the instantaneous radiation is fixed at $I_T = 809.1 W/m^2\ K$, the maximum available for the chosen day. In reality it is necessary to make adjustments and then wait for steady-state conditions between each test point [25].

Water properties are evaluated with the an IAPWS IF-97 code in order to have high accuracy. The heat transfer coefficient inside the tube can be calculated with a simplified procedure that can be retrieved in many heat transfer books (e.g., McAdams, 1954; Kays and Crawford, 1980; Incropera and DeWitt, 2002). The top loss coefficient , back loss coefficient, and overall loss coefficient are found using the methods of Section 6.1.3. The values of F,F' and F_R are found using the methods of Sections 6.1.4 and 6.1.6. The useful energy gain is found using Equation 6.1.35. The wind heat transfer coefficient is found using an approximated correlation of Watmuff et al.(1977)

$$h = 2.8 + 3.0\,V \tag{6.2.19}$$

where $V(m/s)$ is the wind velocity.

A linear fit (using the gross collector area) yields $\eta = 0.84 - 4.49(\Delta T/I)$ and a second order fit yields $\eta = 0.84 - 3.37(\Delta T/I) - 6.19(\Delta T/I)^2$ where $\Delta T = T_{in} - T_a$. The efficiency curve is shown in Figure 60. The efficiency curve is approximated quite perfectly by the second-order polynomial fit although also the linear fit gives a good approximation. The solar collector, for the imposed conditions, presents optimal performances, with really high efficiency for the lowest inlet water temperature. As expected, the efficiency decreases when the inlet water temperature increases. However, The performances shown in Figure 60 assume that the sun is perpendicular to the plane of the collector, which rarely occurs. For the glass cover plates of a typical FPC, specular reflection of radiation occurs, thereby reducing the $(\tau\alpha)$ product [26]. For this reason, it is important to quantify these effects of angle of incident of the incident solar radiation. The incidence angle modifier, $K_{\tau\alpha}$, is defined as the ratio of $(\tau\alpha)$ at some incident angle θ to $(\tau\alpha)$ at normal incidence $(\tau\alpha)_n$. The dependence of $(\tau\alpha)$ on the angle of incidence of radiation on the collector varies from one collector to another, and the standard test methods include experimental estimation of this effect [25].

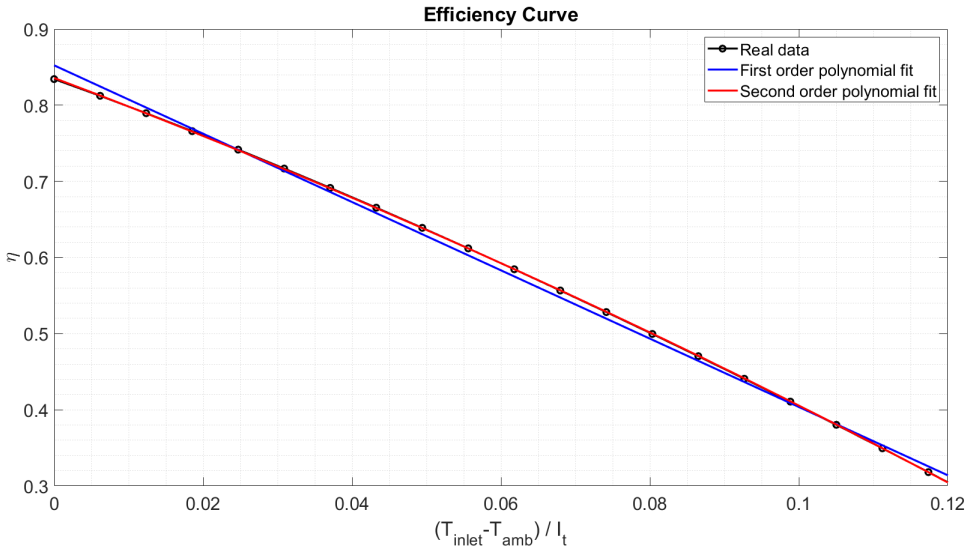


Figure 60: Efficiency curve of the flat-plate solar collector, with fixed conditions

The incident angle modifier for beam radiation incident at angle θ is written as:

$$K_{\tau\alpha}(\theta) = \frac{(\tau\alpha)_b}{(\tau\alpha)_n} \tag{6.2.20}$$

Then for clear days the useful heat is calculated as:

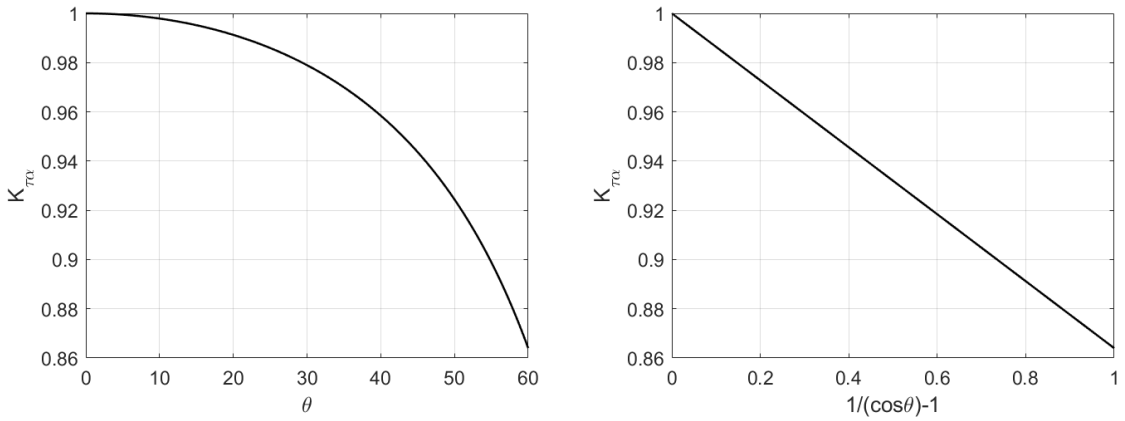
$$Q_u = A_c F_R [I_T K_{\tau\alpha} (\tau\alpha)_n - U_L (T_i - T_a)] \tag{6.2.21}$$

A general expression has been suggested by Souka and Safwat (1966) for angluar dependance of $K_{\tau\alpha}$ for collectors with flat covers as:

$$K_{\tau\alpha} = 1 - b_0 \left(\frac{1}{cos\ \theta} - 1 \right) \tag{6.2.22}$$

where b_0 is constant called the incident angle modifier coefficient. For a single glass cover, the factor b_0 is about 0.1 [26]. For the chosen flat-plate collector, a value of 0.136 is chosen.

In the practice, the test are done with beam incident angles of 0^o, 45^o and 60^o, usually changing the slope of the collector. The problem with changing the slope of this typical flat-plate collector is that changing the slope will impact the convection between the absorber plate and the cover, so that this aspect will be further analysed in the next sections with a numerical simulation.



(a) Dependence on (θ) *(b) Dependence on* $(1/cos\ \theta - 1)$

Figure 61: Incidence angle modifier as function of (θ) *(a) and* $(1/cos\ \theta - 1)$ *for a collector with a flat cover.*

6.3. Numerical simulation for the free convection heat transfer phenomena between absorber and cover

For the design of the performance of solar collectors, the heat losses towards the environment play an important role in determining the efficiency of the collector. It is known that the mayor heat losses are from the top through the glass cover [27], thus the accurate calculation of the heat losses from the top of the collector to the surroundings is very important. Generally, these losses are calculated with approximated heat transfer coefficients, such as the ones for two inclined parallel plates (Duffie and Beckmann, 1991), also used in this work. One of the many important objectives in designing solar collectors is to reduce the heat loss through the covers so that care must be taken. In this last section, a numerical investigation of the natural convection heat transfer in a tilted rectangular, such as the cavity between the absorber plate and the cover of a flat plate solar collector, is presented. A first parametric study is performed varying the aspect ratios $10 \leq A \leq 80$), inclination angles $(15 \leq \beta \leq 45)$ and Rayleigh numbers $(10^4 \leq Ra \leq 10^6)$ of the tilted cavity, in order to validate the model comparing the results to literature. Secondly, an analysis is performed on the exact same collector analyzed with the one-dimensional model in order to compare the *Nu* number obtained and enlighten their differences.

6.3.1. Problem definition

A rectangular cavity with a length L and a height H is shown in Figure 62. Two sidewalls (West and East) are assumed thermally insulated (in reality heat losses through the edges are present), and the other walls are at constant temperatures T_h and T_c, respectively. Air inside the cavity is considered. The Rayleigh numbers are between $10^4 \leq Ra \leq 10^6$, and the aspect ratios of 8, 40 and 80 are considered in the computations. The inclination angles are

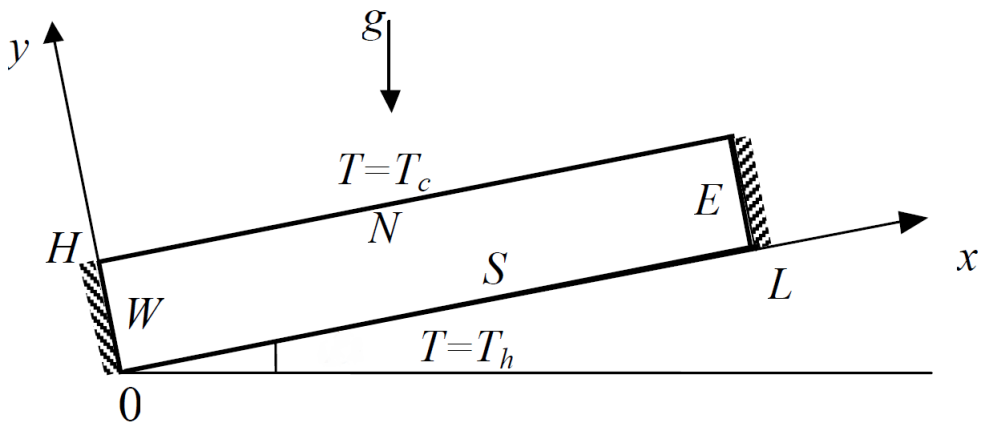


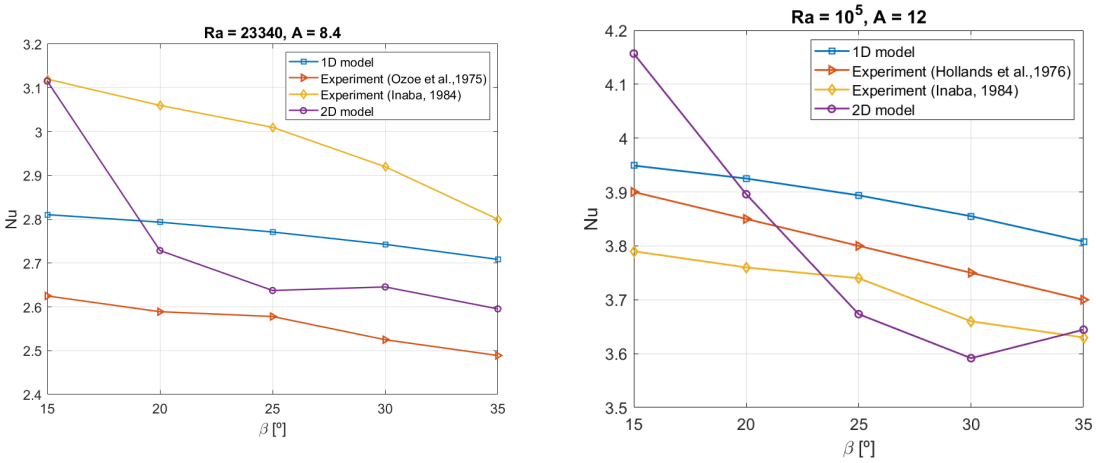
Figure 62: Rectangular cavity physical scheme [27]

in the range of 15°-45°. Others parameters selected are $\Delta T = 50K$,which corresponds to temperature difference between the absorber plate, T_h, and the cover, T_c, in solar collectors.

The assumptions are the same of the DHC problem, except for the fact that the gravity vector now has two components, in spite of one, along the x and y axis. This does not change much the equations of Section 5.3.3 but the discretization procedure has to take into account the change of direction of the acceleration of gravity. As always, three meshes are created to store the data of the scalar values and the velocity field. The mesh are created for each simulation according to the dimension of the cavity and to achieve a computational time of the order of minutes because of the high number of cases studied. As it was made in the DHC cavity study, the Pr number is assumed to remain constant.

6.3.2. Validation of the 2D model

In Figure 63 two test case are simulated, with the exact ratio and Ra of two literature publications [28][29] in order to compare the simulated results with experimental measures. The 1D model is also plotted to show its accuracy. At first glance the two-dimensional model results are similar to the experimental ones but some differences have to be noted. First of all, for these relatively low aspect ratio (A), at $\beta = 15°$ the calculated Nu number is almost always higher then the experimental ones (even though in Figure 63 (a) it is congruent with Inaba results) while when β increases, the Nu shows a descending behaviour similar to the experiments and to the 1D model, although the slope is not monotone but it slightly oscillate. This oscillation could be generated from a slower convergence of the simulation when the slope increases, because the flow becomes more turbulent due to the more important role of the gravity acceleration in modifying the flow field. In order to guarantee the grid of independence of the results, further simulations have been performed for $Ra = 22340,\ 10^5,\ 10^6$ and $A = 8.4$, shown in Figure 64. It can be noted that the results have a strong grid dependence only for $Ra = 10^6$ because, due to flow approaching turbulence, a stable solution is difficult to obtain with a coarse mesh. The evolution of the computed overall Nu over time it is shown



(a) Case study - 1

(b) Case study - 2

Figure 63: Model validation with experimental data with fixed "A" and "Ra"

also in Figure 65 for $A = 8.4$ and $Ra = 23340$ and it can be clearly seen that the oscillations stop after ~ 500 iterations. Nevertheless, the overall results are generally satisfying and show a good accordance with the experimental results, especially considering that the Nusselt number predicted by the 2D model is on the order of the results from experiments and 1D model, while some discrepancies on the trend are observed. These discrepancies may be attributed to aspects such as the presence of thermal radiation, not considered in the 2D model, as well as the fact that the compared experimental results may not fully represent the physics of the flow simulated by the 2D flow, among others. These issues will be addressed in the future works.

In order to pursue a better understanding of the difference of implementing a relatively simple one-dimensional model and a rather more complex two-dimensional one, further simulation are performed with a wide range of geometrical ratio of the cavity while changing also the Ra number and the inclination of the cavity. In particular 36 simulation are performed changing at least one of those parameters, results are shown in Figure 66 and Table 21. In the table are shown also the used mesh for each simulation. In order to have a reasonable computational time, the total number of CVs was take similar to the ones used in the simulation of the DHC problem (~ 10000). The mesh has been changed along with ratio of the cavity. For high ratio the length L of the cavity has to increase a lot in order to achieve the desired Ra so it is natural to increase the number of CVs in the direction of L; nevertheless, the interest is in the phenomena that happen along the H direction of the cavity, thus it was preferred to maintain, also for high ratio, a significant number of CVs, although H was only a few centimeters long.

The overall results are satisfying and they do not show any important difference, but there are some general observation that can be made looking at the figures. For low geometrical ratio A, the second-dimensional model, particularly for $Ra = 10^6$, tends to underestimate the Nu respect to the correlation. For $A = 40$ and for $Ra = 10^5$ and $Ra = 10^6$ the simulated Nu is

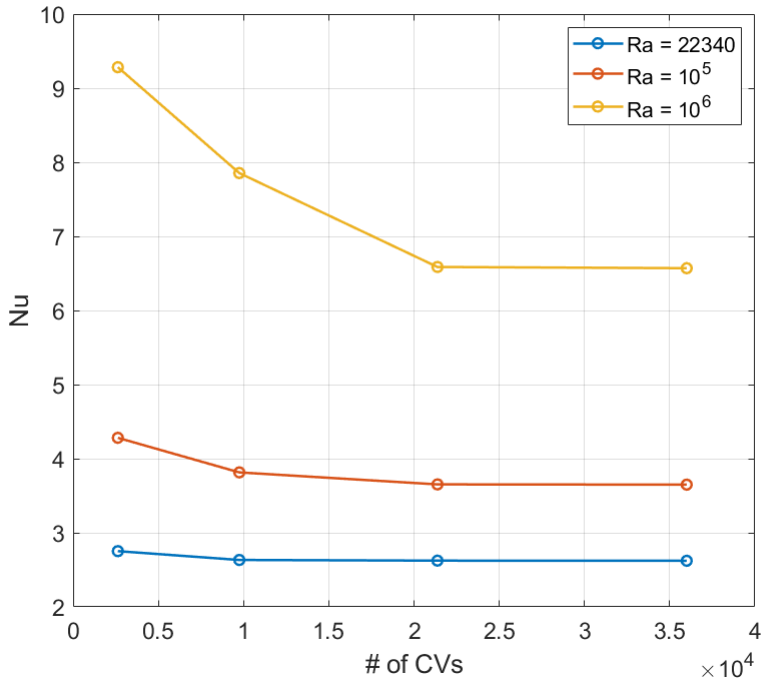


Figure 64: Grid Independence for several Ra and A = 8.4

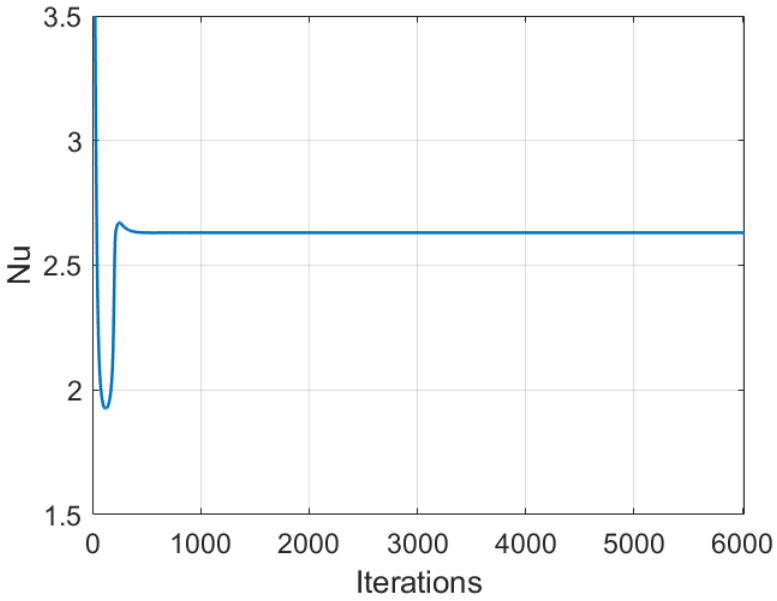
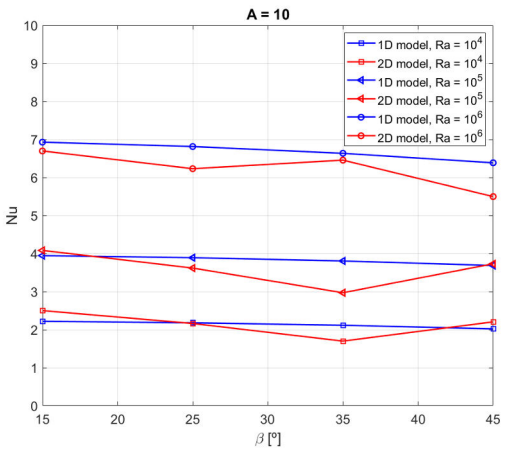


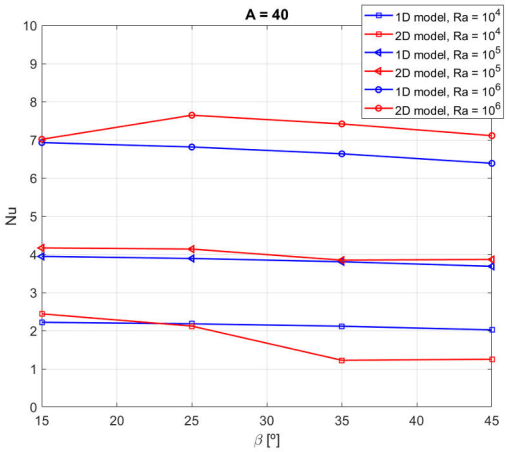
Figure 65: Global Nu for Ra = 23340 and A = 8.4

Ratio (A)	**Mesh**	**Ra**	β	$\mathbf{Nu_{corr}}$	$\mathbf{Nu_{2D}}$
10	150x65	10^4	15º	2.219	2.501
			25º	2.179	2.162
			35º	2.116	1.699
			45º	2.020	2.205
		10^5	15º	3.946	4.083
			25º	3.891	3.619
			35º	3.805	2.970
			45º	3.685	3.730
		10^6	15º	6.928	6.696
			25º	6.812	6.230
			35º	6.634	6.455
			45º	6.385	5.497
40	170x45	10^4	15º	2.219	2.439
			25º	2.179	2.119
			35º	2.116	1.224
			45º	2.020	1.251
		10^5	15º	3.946	4.167
			25º	3.891	4.137
			35º	3.805	3.847
			45º	3.685	3.865
		10^6	15º	6.928	7.014
			25º	6.812	7.645
			35º	6.634	7.417
			45º	6.385	7.109
80	180x35	10^4	15º	2.219	2.575
			25º	2.179	2.199
			35º	2.116	1.175
			45º	2.020	1.213
		10^5	15º	3.946	4.315
			25º	3.891	4.163
			35º	3.805	3.906
			45º	3.685	3.421
		10^6	15º	6.928	7.687
			25º	6.812	7.811
			35º	6.634	6.760
			45º	6.385	6.641

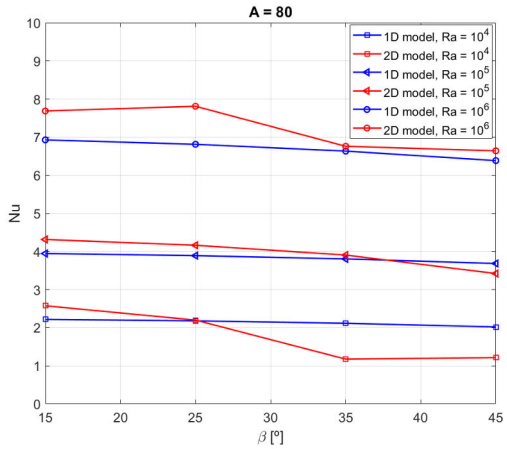
Table 21: Nu calculated with the 2D simulation compared with 1D correlation



(a) Parametric study with $A = 10$



(b) Parametric study with $A = 40$



(c) Parametric study with $A = 80$

Figure 66: Comparison between 1D correlation and 2D simulation of the average Nusselt number inside a rectangular inclinated cavity

now overestimated respect to the correlation while for $Ra = 10^4$ the Nu continues to be underestimated. The situation for high geometrical ratios $A = 80$ does not change significantly, the Nusselt is overestimated at high Rayleigh numbers while it is underestimated at low Ra. It has to be noted that the 1D correlation does not take into account the geometrical ratio and the major discrepancies (if the 2D simulation is assumed more accurate) are found for $Ra = 10^6$, probably due to the fact that the flow is approaching the turbulence and the correlation is not accurate enough to take into account the phenomenon. Moreover, the effect of the β on the average Nusselt number is almost negligible if compared to the effect of the Ra number, so that the inclination of a solar collector should be designed looking at maximizing the impacting solar radiation rather than the minimizing the heat thermal losses through the environment.

A	89.3
Ra	$1.48 \cdot 10^5$
β	15.3

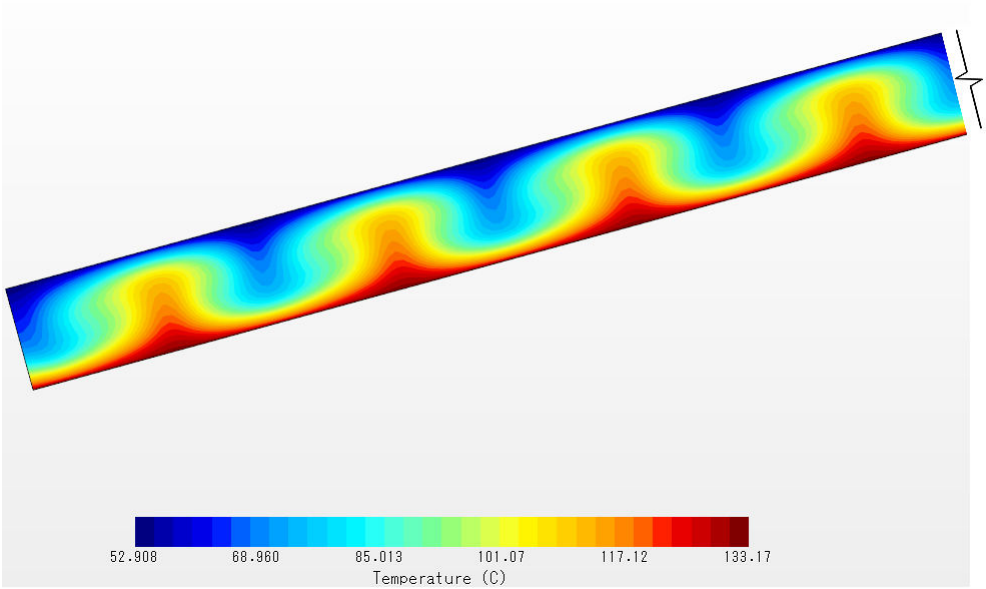
Table 22: Parameter simulation for a "real" flat-plate solar collector

6.3.3. Two-dimensional simulation results and comments

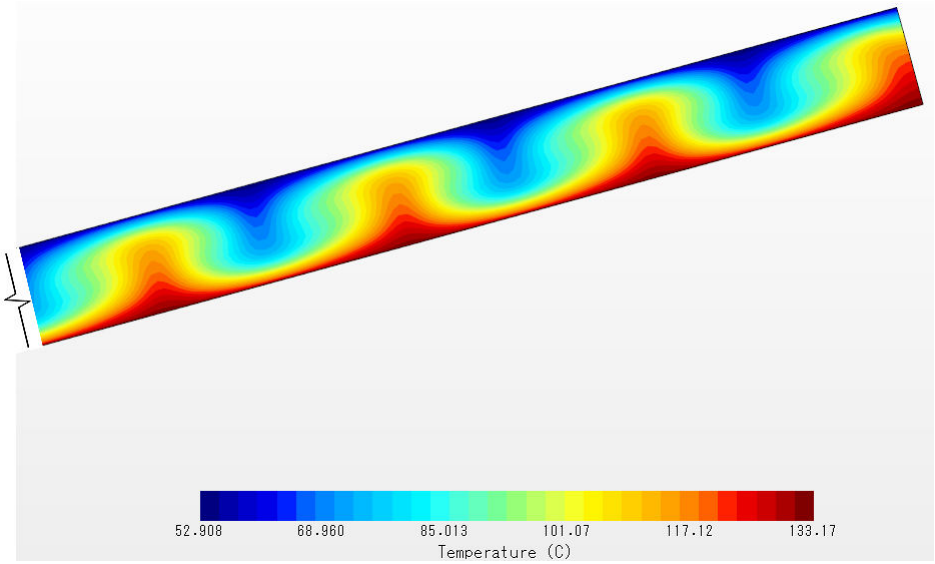
A flat-plate solar collector with the same exact specification of Section 6.2.3 is simulated in order to have a direct comparison with the two models. The simulation parameters for this simulation are shown in Table 22.

The resulting Nusselt calculated with the simulation is $Nu_{sim} = 4.19$ while the one calculated with the correlation is $Nu_{corr} = 3.91$. The simulation estimates a higher Nusselt number as expected with the considerations explained in Section 6.3.2. The results are shown in Figures 67 to 70. It can be noted from these figures that all the presented results show a repeating pattern along the length of the collector, for this reason not all the simulation domain is presented in favour of only two sections (left and right sides) of the cavity. All the contour plot patterns have a characteristic length of $\sim 1.5 \div 2$ times the distance from the absorber and the cover. It can be seen in Figure 67 that the temperature is higher in each "pattern section" center, where heat is transported mainly by advection. Indeed, highest temperature regions are characterized by highest vertical velocities. Figure 70 shows the creations of small vortex all along the collector cavity, it is clearly visible that two different vortexes are alternating when moving along the cavity, one of them presents higher velocity vectors while the other one has less intensity. All the velocity field plots recall the DHC cavity plots of Figure 50 and 51 at $Ra = 10^4$ and same comments could be made about the slope of the velocity components contour plots. It is interesting that different "small" vortexes are created instead of only a "big" one; it can be explained by the fact that the inclination of the cavity is not elevated ($\beta = 15.3^o$) and that the distance between the absorber plate and the cover is little if compared to the collector length. It was found from the simulations of Section 6.3.2 that the vortexes tend to stretch out when increasing the β in a similar way as shown in Figure 71, causing a slight reduction of the Nu and the overall loss to the ambient.

Regarding the matter of validating the code, further work is surely needed. In this work, the code applied to this rectangular cavity was only validated properly for a square cavity with $\beta = 0$ in Section 5.3. The verification that has been done for the square cavity, however, is not enough to ensure the validity of the code. Although a comparison of the average Nusselt obtained with the results of two experiments and a correlation has been performed, correct validation requires an accuracy check over much more data (again, as in Section 5.3.4). Furthermore, the calculated Nusselt is an average value, not punctual and therefore not suitable for such a study. A future approach could be to find in the literature a benchmark

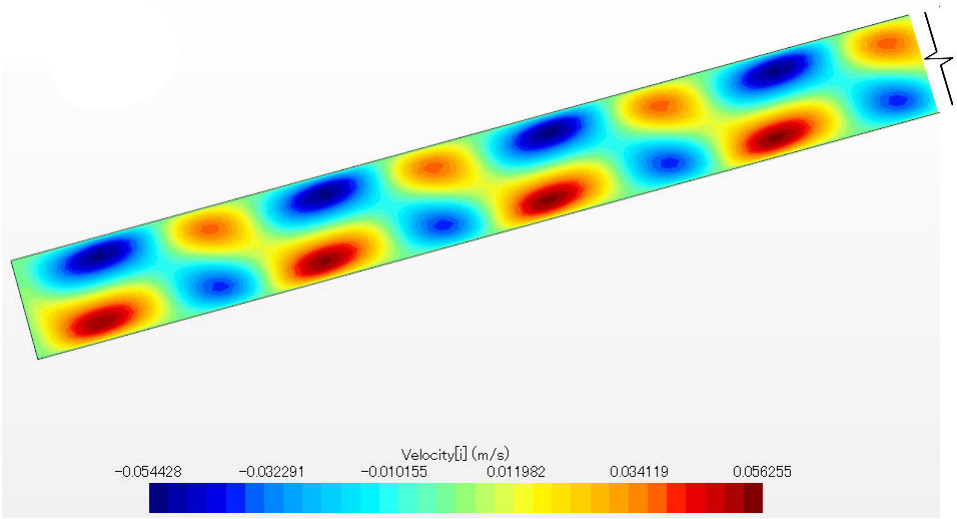


(a) Left side

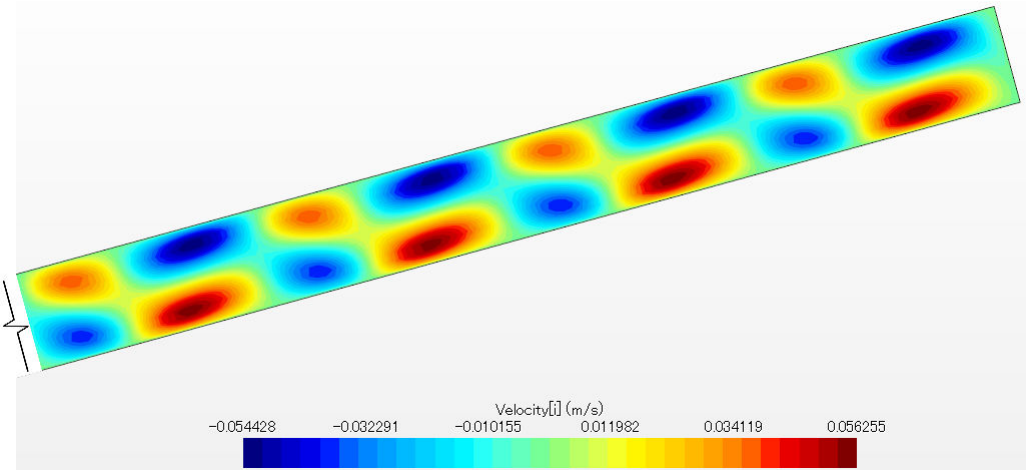


(b) Right side

Figure 67: Temperature contour plots

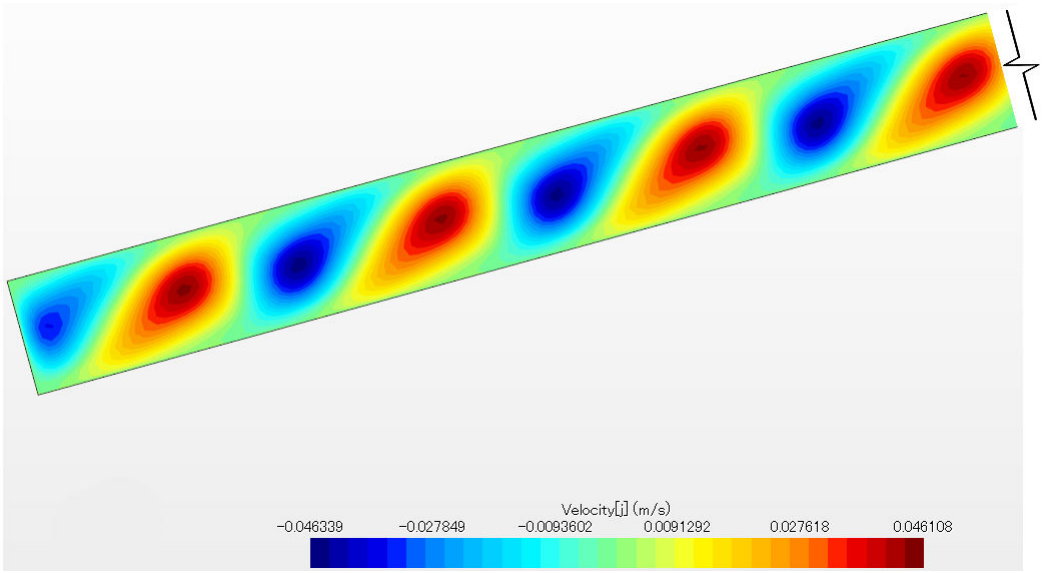


(a) Left side

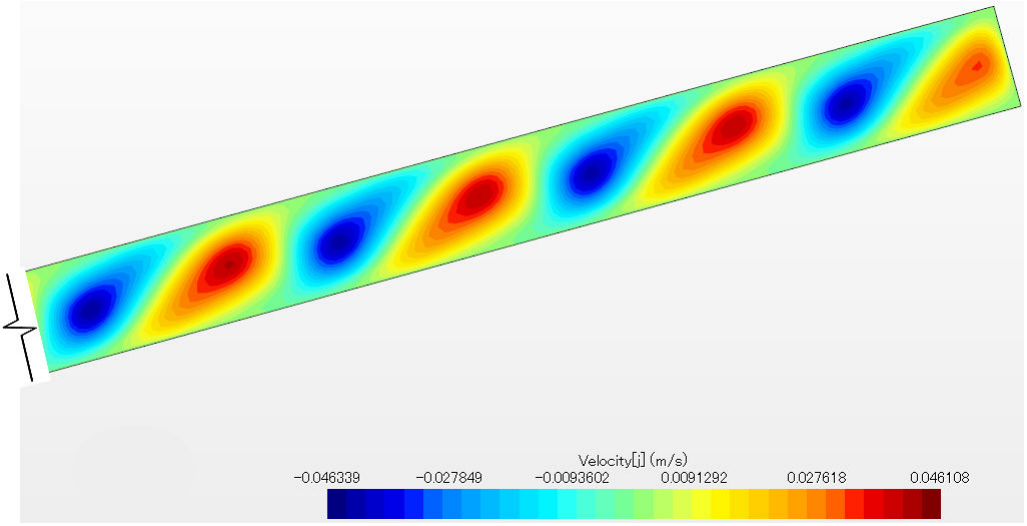


(b) Right side

Figure 68: Horizontal velocity component u contour plots

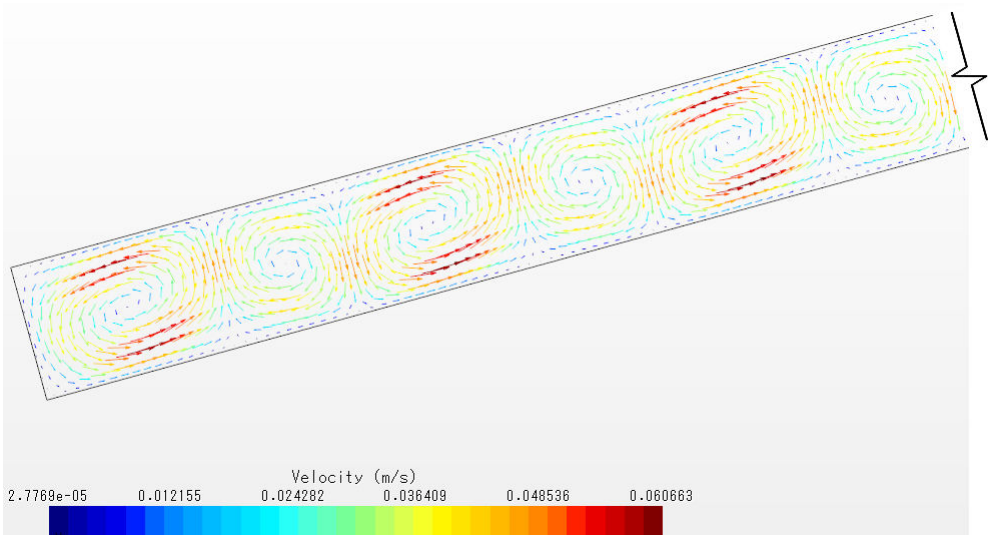


(a) Left side

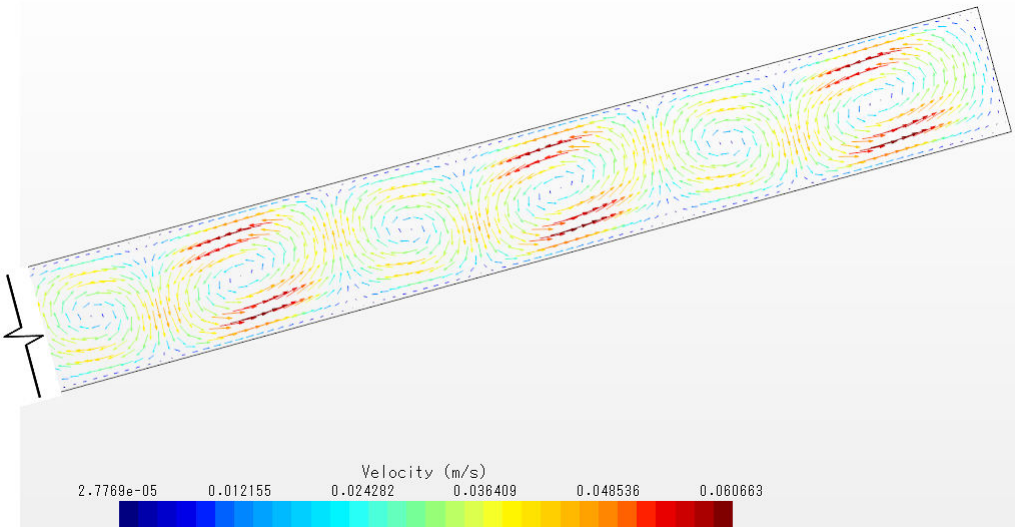


(b) Right side

Figure 69: Vertical velocity component v contour plots



(a) Left side



(b) Right side

Figure 70: velocity field vector plot

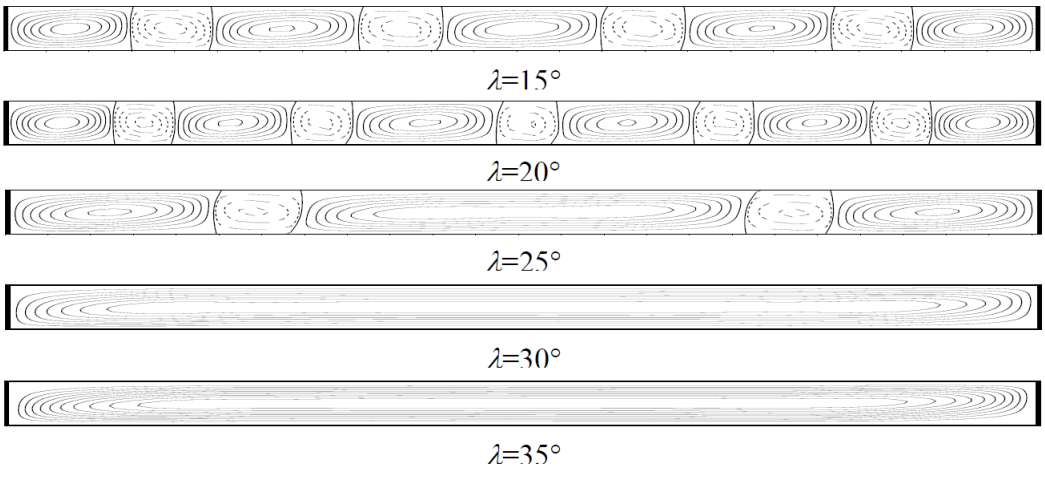


Figure 71: Streamlines for a cavity of $A = 16$, $Ra = 10^4$ for $\beta = 15^o - 35^o$[27]

case with all the necessary data or to verify the code with an experimental setup, with a similar collector installed in the same place and under the same conditions.

7. Conclusions and possible future research

This study was intended to provide the author of this paper with expertise in the field of numerical methods applied to Heat Transfer and Computational Fluid Dynamics. Thanks to the development of this work, a general knowledge of the methods and numerical algorithms used to solve complex equations such Navier-Stokes was acquired.

In general terms, the scope and the of the study has been accomplished. The first part of the study, the development of the numerical methods and the resolution of benchmark problems, has produced satisfying results, since the developed codes solutions coincided with benchmark solutions with a reasonable accuracy, with slight discrepancies due to the different numerical methods and approximations that were used or to the different types and refinements of meshes.

One of the most remarkable achievements of this first study is the fact that one learns how to obtain good solutions with a desired order of accuracy. Moreover, the ability to discuss a solution has been acquired. When applying a numerical method to some physical phenomenon, it is always necessary to check that the solution makes sense from both a numerical point of view (e.g. with a grid independence) but also from a physical point of view. For this reason, in all sections of the project there is, first of all, a verification that the code developed works correctly and then an analysis of the phenomenon from a physical point of view, trying to provide an explanation of the phenomenon analysed and why the solution obtained can be considered reliable.

In the final part of the work, these numerical methods were applied to a study of a flat-plate solar collector. A brief introduction on the operation of this type of collectors was provided, together with a one-dimensional model to analyse the performance of the collector. The study of natural convection in a cavity has therefore been applied to the study of heat transfer between the absorber plate and the cover, allowing a comparison with the one-dimensional model. Although the results of this last analysis have been in line with some experiments found in literature, a validation must be further deepened with an accurate research of a benchmark case in literature or with a measurement and an experimental study of a collector operating in the same conditions and in the same geographical position. Nevertheless, it has been demonstrated that HF and CFD codes can be used (and are often used) in many energy applications and can prove to be a truly versatile and powerful tool in the hands of an energy engineer.

Regarding the future research, surely an experimental study of a flat-plate collector should be developed and used as a benchmark setup for the validation of CFD and HT codes. The next step can be to expand the area of validity of these codes in the phenomenon of turbulence. Turbulence is present in many practical applications and consists of a real step forward in the development of CFD codes. A further improvement in the code would certainly be to move from a two-dimensional domain to a real three-dimensional domain, thus opening

up the possibility of simulating phenomena, both in turbulent nature and in geometries that cannot be simplified into two or one dimension. Once the code developed has been found to be satisfactory, the possible applications are countless. One possibility is the accurate simulation of energy production processes using renewable sources, such as flat-plate solar collectors or concentrating solar collectors, which are widely used in Spain and which certainly need a high level of optimisation for their use to be profitable.

Bibliography

[1] Carlone, P., Aleksendric, D. (2015). *Soft Computing in the Design and Manufacturing of Composite Materials: Applications to Brake Friction and Thermoset Matrix Composites.* Woodhead Publishing.

[2] Patankar, S. V. (1980). *Numerical heat transfer and fluid flow*. Hemisphere Publishing Corporation (CRC Press, Taylor Francis Group). ISBN: 978-0891165224

[3] Quarteroni, A. Saleri, F. (2006).*Scientific Computing with MATLAB and Octave*. Berlin: Springer-Verlag.

[4] Direct and iterative method. (2020). Retrieved 26 June 2020, from `http://www.ukessays.com/essays/mathematics/direct-and-iterative-method.php`

[5] Becker, T. W. and Kaus, B. J. P (2016): *Numerical Modeling of Earth Systems: An introduction to computational methods with focus on solid Earth applications of continuum mechanics*, University of Southern California, v. 1.2, 2016

[6] Thermal diffusivity. (2020). Retrieved 1 July 2020, from `https://en.wikipedia.org/wiki/Thermal_diffusivity`

[7] B.P. Leonard and Simin Mokhtari. *Ultra-Sharp solution of the Smith-Hutton Problem*. Insitute for Computational Mechanics in propulsion Lewis Research Center, Cleveland, Ohio 44135.

[8] F. Moukalled and M. Darwish. *A Comparative Assessment of the Performance of Mass Conservative-Based Algorithms for Incompressible Multiphase Flows*. Numerical Heat Transfer, Part B, 42:259–283, 2002.

[9] Montllor Ramoneda, Oliva Llena, Perez Segarra. *Computational Fluid Dynamics: Fractional Step Method and its applications in internal and external flows*. June, 10th 2018. ESCOLA SUPERIOR D'ENGINYERIES INDUSTRIAL

[10] Raithby, G.D. and Hollands, K.G.T., 1975, *A General Method of Obtaining Approximate Solutions to Laminar and Turbulent Free Convection Problems*, Advances in Heat Transfer, eds. T.F. Irvine Jr. and J.P. Hartnett, Academic Press, New York, pp.265-315.

[11] Chorin, A. (1968). *Numerical solution of the Navier-Stokes equations*. Mathematics Of Computation, 22(104), 745-745. doi: 10.1090/s0025-5718-1968-0242392-2

[12] Ferziger, J., Peric, M., Street, R (2002). *Computational methods for fluid dynamics*.

[13] Courant, R.; Friedrichs, K.; Lewy, H. *Über die partiellen Differenzengleichungen der mathematischen Physik*. (German) Math. Ann. 100 (1928), no. 1, 32–74.

[14] CTTC, Centre Tecnològic de Transferència de Calor,*Introduction to the Fractional-Step Method*, Universitat Politècnica de Catalunya, 2017.

[15] Erwin Simons. An efficient multi-domain approach to large eddy simulation of incompressible turbulent flows in complex geometries. PhD thesis, Von Karman Institute for Fluid Dynamics, October 2000.

[16] Course on Numerical Methods in Heat Transfer and Fluid Dynamics, ETSEIB

[17] U. Ghia, K. N. Ghia, and C. T. Shin. High-Re solutions for incompressible flow using the Navier-Stokes equations and a multigrid method. In: Journal of Computational Physics 48.3 (1982), pp. 387–411. issn: 10902716. doi: 10.1016/0021- 9991(82)90058-4. arXiv: arXiv:1011.1669v3.

[18] Lid-driven cavity problem - CFD-Wiki, the free CFD reference. (2020). Retrieved 15 August 2020, from `https://www.cfd-online.com/Wiki/Lid-driven_cavity_problem`

[19] Pressure equation method, Staggered grid system (2020). Retrieved 17 August 2020, from `https://www.youtube.com/watch?v=4iuHCrDVxkM`

[20] Munson, B. (2013). *Fundamentals of fluid mechanics*. Hoboken, N.J: Wiley.

[21] De Vahl Davis, G. (1983). *Natural convection of air in a square cavity: A bench mark numerical solution*. International Journal For Numerical Methods In Fluids, 3(3), 249-264. doi: 10.1002/fld.1650030305

[22] Prandtl Number. (2020). Retrieved 24 August 2020, from `http://thermopedia.com/content/1053`

[23] Rayleigh number. (2020). Retrieved 24 August 2020, from `https://en.wikipedia.org/wiki/Rayleigh_number`

[24] Nusselt number. (2020). Retrieved 26 August 2020, from `https://en.wikipedia.org/wiki/Nusselt_number`

[25] Duffie, J., Beckman, W. (1980). *Solar engineering of thermal processes*. New York: Wiley.

[26] Kalogirou, S. (2014). *Solar energy engineering*. Amsterdam: Academic Press.

[27] Álvarez-García, G., Xamán, J., Jassón, F., Alvarado Juárez, R. (2007).*Nusselt Number for the Natural Convection and Surface Thermal Radiation in Solar Collectors*.

[28] I. Inaba, *Experimental study of natural convection in an inclined air layer*, Int. J. Heat Mass Transfer 27, 1127- 1139, 1984.

[29] K.G.T. Hollands, T.E. Unny, G.D. Raithby, L. Konicek, *Free convective heat transfer across inclined air layers*. J. Heat Transfer 98, 189-192, 1976.