Study for the computational resolution of conservation equations of mass, momentum and energy.

Student name: Altadill Llasat, Miquel

Director name: Perez Segarra, Carlos David

Co-Director name: Oliva Llena, Asensio

Department: Centre Tecnològic de Transferència de Calor (CTTC)

Document: Final Report

Delivery Date: 10/06/2019

Aerospace Vehicles Engineering - ESEIAAT
Universitat Politècnica de Catalunya (UPC)

Agradecimientos

Resumen

El objetivo de este proyecto es el estudio de las equaciones de la conservación de la massa, el momento y la energia para posteriormente desarroyar un codigo no comercial en lenguaje C y Matlab. Este software ha de ser capaz de resolver las equaciones de Navier-Stokes por lo que serà necessaria una buena comprensión tanto matematica como fisica de estas. Aunque por su complejidad serà necessario realizar casos con geometrias sencillas haviendo supuesto antes las hipotesis necesarias.

El proposito principal de este estudio es poder llegar a realizar simulaciones validas del comportamiento termico y fluido-dinamico de casos estudiados previamente. Los casos principalmente estudiados són un problema de transmission de calor por conduccion bidimensional con diferentes materiales y el problema de Smith-Hutton. Al ser este un primer acercamiento a este campo de estudio, ha sido buena idea escojes problemas ya estudiados, facilitando así el dessarroyo de este trabajo.

Abstract

El objetivo de este proyecto es el estudio de las equaciones de la conservación de la massa, el momento y la energia para posteriormente desarroyar un codigo no comercial en lenguaje C y Matlab. Este software ha de ser capaz de resolver las equaciones de Navier-Stokes por lo que serà necessaria una buena comprensión tanto matematica como fisica de estas. Aunque por su complejidad serà necessario realizar casos con geometrias sencillas haviendo supuesto antes las hipotesis necesarias.

El proposito principal de este estudio es poder llegar a realizar simulaciones validas del comportamiento termico y fluido-dinamico de casos estudiados previamente. Los casos principalmente estudiados són un problema de transmission de calor por conduccion bidimensional con diferentes materiales y el problema de Smith-Hutton. Al ser este un primer acercamiento a este campo de estudio, ha sido buena idea escojes problemas ya estudiados, facilitando así el dessarroyo de este trabajo.

Contents

List of Symbols

t	Time. $[s]$
ϕ	General Variable.
T	Temperature. $[^{\circ}C]$
T_I	Temperature at Grid-Point I. $[^{\circ}C]$
k	Conductivity. $[W/mK]$
k_i	Conductivity at face point i.
$(\partial x)_i$	Distance between grid-points. $[m]$
G	Source term.
$\overline{G}$	Average rate of heat generation in the CV.
Δx	Control volume length in direction X. $[m]$
ρ	Density. $[kg/m^3]$
c_p	Specific heat. $[J/kgK]$
v	Velocity in X axis. $[m/s]$
u	Velocity in Y axis. $[m/s]$
w	Velocity in Z axis. $[m/s]$
p	Pressure. $[Pa]$
g	Gravity. $[m/s^2]$
$\overrightarrow{\tau}$	Shear stress tensor. $[N/m^2]$
$\overrightarrow{q}$	Heat flow.
u	Internal energy in N-S equations $[J]$
e_c	Kinetic energy in N-S equations $[J]$
μ	Dynamic viscosity $[Ns/m^2]$

J_i	Total face flux term in direction i
F_i	Mass flow rate through CV face i
Pe	Peclet Number $\rho u \delta x / \Gamma$
k	Current iteration count
N_x	Number of nodes in direction X
N_y	Number of nodes in direction Y

Acronyms

1D One-Dimensional. 27, 30, 31, 39, 43

2D Two-Dimensional. 23, 27, 31, 42–44

3D Three-Dimensional. 31, 43

CDS Central Difference Scheme. 27–29

CV Control Volume. 5, 22, 23, 25–28

EDS Exponential Difference Scheme. 27, 30, 31

FDM Finite Difference Method. 3, 33–35

FEM Finite Element Method. 34

FVM Finite Volume Method. 3, 25, 33–35

HDS Hybrid Difference Scheme. 27, 29, 31

LU Lower-Upper decomposition. 39

N-S Navier-Stokes. 5, 9, 21–23, 25

PDE Partial Differential Equations. 35

PLDS Powerlaw Difference Scheme. 27, 31

TDMA Tri-Diagonal Matrix Algorithm. 39, 40, 43

UDS Upwind Difference Scheme. 27–29

List of Figures

List of Tables

Chapter 1

Introduction

Contents

1.1 Aim

The aim of this project is to study basic integration cases of mass, momentum and energy conservation equations for consolidating the knowledge about this thematic as well as the mathematical formulations, computational and programming techniques, etc. During study we will need to study some previous programming examples before we can start coding the core of our project in order to learn and practice the basic skills needed for accomplishing the final objective of this project, run a simulation focused on specific applications in the field of study about optimization of thermo-fluidic and aeronautical systems for maximizing their energetic efficiency as well as other computational aspects and reducing their costs and environmental impact.

1.2 Scope of Study

In this section it is shown a list of tasks that composes the core of the study. From the beginning of the project it has been necessary to follow some guidelines and calendars in order to accomplish the the tasks listed below:

- Previous Background analysis and state-of-the-art on numerical simulations in the field of fluid dynamics and heat and mass transfer.
- Approach to the physical phenomenon and mathematical formulation.
- Development of the numerical simulation tools required in C language and Matlab environment.
- Code and numerical solutions verification and validation from experimental data or simulations made by researchers.
- Obtaining and analyzing the results from the simulations in order to extract some conclusions.
- Application to the specific case selected if possible.
- Analysis of possible design optimization of the selected case.
- Conclusions and redaction of the final report.

1.3 Requirements

The requirements taken into account in this study are listed below. Some of them have been selected since the beginning of the project but some have been introduced as it progressed in order to accomplish the tasks listed in the previous section.

- Programming language: C and Matlab
- Developement of an own non-commercial code
- Data visualization program: Matlab and Excel.

- The cases of study and domain must be Two-Dimensional
- The mesh used for the domain discretization will be structured.The justification is shown in Section **??**
- The developed code must be able to solve steady and non-steady state cases
- Navier-Stokes Equations need to be solved with certain assumptions and simplifications. It is shown in Section 2.3.2.
- Constant physical properties assumed for all the cases but the code could run with variable properties if some modifications are done.
- Newtonian fluid assumed.
- For the Convection-Diffusion equation study

1.4 Justification

1.5 State of the Art

Chapter 2

Approach to the physical phenomenon and mathematical formulation

Contents

Many problems that involves the resolution of differential equations can be solved *analytically* specially those ones that involve simple geometries with simple boundary conditions. But when the problem involve complicated geometries with complex *boundary conditions* and variable properties its needed another method for solving the equations involved in the physical phenomenon. For this cases we can still obtain sufficiently accurate approximate solutions using *numerical methods*, those are based on replacing the differential equation by a set of n algebraic equations for the unknown medium property at n selected points of the medium, and the simultaneous solution of these equation results in the medium property values at those *discrete points*. We are going to call this arbitrary medium property or dependent variable ϕ to refer to it in the following sections. [3]

The numerical solution of heat transfer, fluid flow, and other related processes can begin when the laws governing these processes have been expressed in mathematical form, generally in terms of differential equations. In this section we are going to develop the mathematical formulation and complete derivation of these equations as an initial step for developing the code for modelling these phenomenons [1]. The purpose in this section is to develop the familiarity with the form and meaning of these equations, geometric formulation of the control volume and the main ingredients for developing the *numerical simulation* tools for each case of study.

2.1 Control-Volume Formulation

The basic idea of the control-volume formulation lends itself to direct physical interpretation. The calculation domain is divided into a number of nonoverlapping control volumes such that there is one control volume surrounding each grid point. Then we have to integrate the differential equation over each control volume. Piecewise profile seen in Fig. 2.2b shows the variation of ϕ between their neighbour grid points used for evaluating the govern equations integrals resulting in a discretization equation that will contain the values of ϕ for a group of grid points.

The conservation equation obtained from the finite control volume study express his conservation principles for ϕ, just as the differential equation express it for an infinitesimal control volume. One of the most desirable features of this formulation is that the resulting solution would imply the integral conservation quantities such as mass, momentum and energy is exactly satisfied over any group of control volumes and over the whole calculation domain.

When we want to solve the discretization equation to obtain the grid-point values of ϕ we only need this values to constitute the solution, without any explicit reference as to how ϕ varies between the grid points. The solution should only be dependent only on the grid point values. Once defined the numerical method formulation we can start with a simple case formulation such as one-dimensional heat conduction for a better comprehension.

2.2 Heat Conduction

"Conduction is the transfer of energy from more energetic particles of a substance to the adjacent less energetic ones as a result of interactions between particles"[3]. This phenomenon can take place in solids due to the combination of vibrations of the molecules in lattice and the energy transport by free electron and in liquids or gases it is caused by the collisions and *diffusion* of the molecules during their random motion. In this section we are going to make the numerical formulation for *heat conduction* in solids.

There are several ways of obtaining the numerical formulation of a heat conduction problem, such as the *finite difference* method, the *finite element* method, the *boundary element*, and the *energy balance* (or control volume method). Each method has its own advantages and disadvantages, and each of them has their applications in various fields of study. In the Section 3.2 we can find a description for each method.

We are going to use the *energy balance* approach for a better comprehension of the phenomenon since it is based on the familiar energy balances on control volumes instead of heavy mathematical formulations, and thus gives a better feel for the problem. Besides, it results in the same set of algebraic equations as the *finite difference* method. It will be first demonstrated heat conduction for steady state problems and then for transient cases.

2.2.1 Steady One-Dimensional Conduction

The govern equation found via heat balance in Appendix A for steady one-dimensional heat conduction is

$$\frac{d}{dx}\left(k\frac{dT}{dx}\right) + G = 0 \tag{2.1}$$

were T is the temperature, k is the conductivity and G is the rate of heat generation per unit volume.

To derive the discretization equation we will employ the grid-point scheme shown in the Fig.2.1. This scheme can be extended to two and three-dimensional situations that is why we will only describe it in this section. Grid point P has grid)points W and E as neighbours, E denotes east $(+x)$ side and W west side $(-x)$. The dashed lines show the faces of the control volume denoted by letters e and w following the previous criterion.

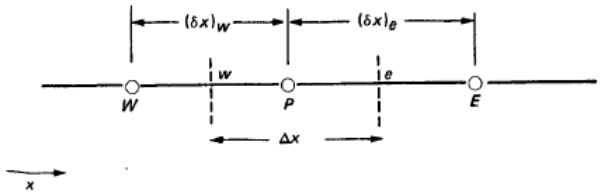


Figure 2.1: Grid-point scheme for one-dimensional problem[1]

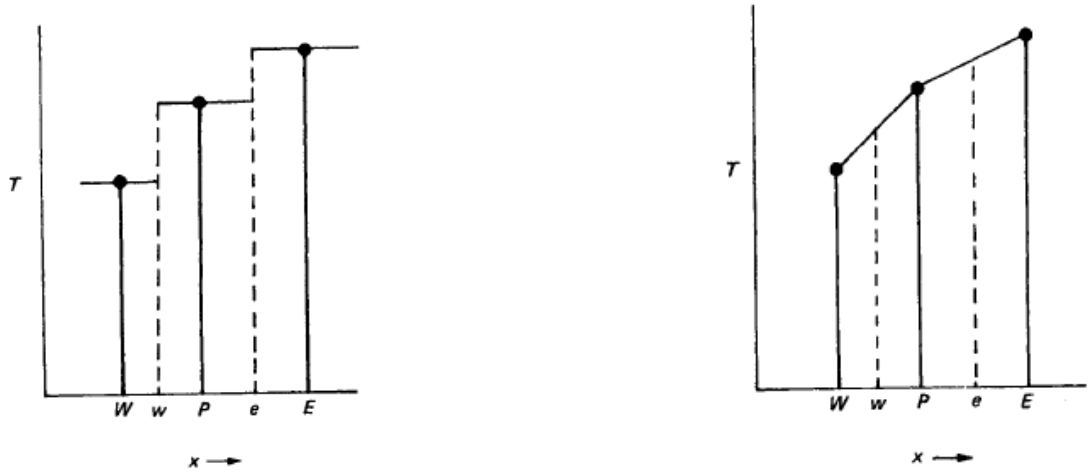
Integrating over the control volume, we get

$$\left(k\frac{dT}{dx}\right)_e - \left(k\frac{dT}{dx}\right)_w + \int_w^e G dx = 0 \tag{2.2}$$

Now it is the moment when we need a profile assumption or an interpolation formula for expressing the derivative terms of Eq.2.2. We have two possibilities, the simplest one is to assume that the value of T at a grid point remains constant over the control-volume faces. This is called stepwise profile and its shown in Fig.2.2a and we should notice that for this profile we can not define the slope dT/dx at the control volume faces. The other possibility is the piecewise-linear profile (Fig.2.2b) that does a linear interpolation between the grid points avoiding the previous profile problem. Evaluating the derivatives dT/dx in Eq.2.2 from the piecewise-linear profile we obtain

$$\frac{k_e(T_E - T_P)}{(\partial x)_e} + \frac{k_w(T_W - T_P)}{(\partial x)_w} + \overline{G}\Delta x = 0 \tag{2.3}$$

where T_E, T_W and T_P stands for the temperature at the grid-point, $\overline{G}$ is the average rate of heat generation in the control volume, k_e and k_w represents the the conductivity over each control volume boundary and $(\partial x)_e$, $(\partial x)_w$ and Δx are related to the grid-point geometry shown in Fig.2.1 .



(a) (b)

Figure 2.2: Two profile assumptions (a)Stepwise profile (b)Piecewise-linear profile[1]

Arranging the terms from Eq.2.3

$$\frac{k_e}{(\partial x)_e}T_E + \frac{k_w}{(\partial x)_w}T_W - T_P\Big(\frac{k_e}{(\partial x)_e} + \frac{k_w}{(\partial x)_w}\Big) + \overline{G}\Delta x = 0 \tag{2.4}$$

we obtain Eq.2.4 which can be rearranged into the following form:

$$a_P T_P = a_E T_E + a_W T_W + b \tag{2.5}$$

where

$$a_E = \frac{k_e}{(\partial x)_e} \tag{2.6a}$$

$$a_W = \frac{k_w}{(\partial x)_w} \tag{2.6b}$$

$$a_P = a_E + a_W \tag{2.6c}$$

$$b = \overline{G}\Delta x \tag{2.6d}$$

Equation 2.8 represents the standard form in which we will write our further discretization equations. We arranged this equation arranging the temperature of the center grid point T_P on the left side of the equation, leaving the neighbour point temperatures and their coefficients to the right side with the constant b.

It is important to understand the treatment of the source term (G) before we proceed to develop the solver. Often the source term is a function of the dependent variable T itself, this dependence affects to the construction of the discretization equation. As they will be solved by the techniques for linear algebraic equations we can linearize the relation given by $G \sim T$ [1] and express the average value $\overline{G}$ as

$$\overline{G} = G_c + G_P T_P \tag{2.7}$$

With the linearized source expression the discretization equation would still look like Eq.2.8, but the coefficient definitions would change:

$$a_P T_P = a_E T_E + a_W T_W + b \tag{2.8}$$

where

$$a_E = \frac{k_e}{(\partial x)_e} \tag{2.9a}$$

$$a_W = \frac{k_w}{(\partial x)_w} \tag{2.9b}$$

$$a_P = a_E + a_W - G_P \Delta x \tag{2.9c}$$

$$b = G_C \Delta x \tag{2.9d}$$

It is important to understand that we need not use the same profile assumption for all quantities. $\overline{G}$ need not be calculated from a linear variation of G between the grid points but the assumptions done leads to simpler discretization equation formulation[1]. This introductory formulation provides sufficient background to allow the formulation rules that our discretization equations should obey to ensure physical realism and overall balance for multi-dimensional heat conduction problems.

Interface Conductivity

Before jumping into the unsteady analysis it is a good idea to present the interface conductivity concept. As we said, k_e and k_w represents the the conductivity over each control volume boundary. In some cases the conductivity is a function of x (nonuniform), we shall know the values of k only at the grid points W, P, E and others for multi-dimensional situations.

Nonuniform conductivity can arise from nonhomogenity of the material as in a composite slab, as we will see in the first solver. Even in a homogeneous material the temperature dependence of conductivity can lead to conductivity variations in response to the temperature distribution. One procedure for finding the interface conductivity is to assume a linear variation of k between points P and E making reference to Fig.C.1:

$$k_e = f_e k_P + (1 - f_e) k_E \tag{2.10}$$

where f_e is the interpolation defined by:

$$f_e = \frac{(\partial x)_{e+}}{(\partial x)_e} \tag{2.11}$$

[1] Procedure shown in [1]

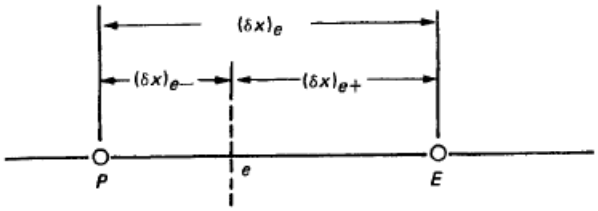


Figure 2.3: Geometry of the Control Volume interface e [1]

As we can see in Fig.C.1 the interface is not located midway between the points P and E that is because we want to study the most generic case and once its done we will focus into our problem conditions. One alternative for obtaining the interface conductivity is to obtain a good representation of the heat flux q_e over interface e. As we know:

$$q_e = k_e \frac{dT}{dX} = \frac{k_e(T_P - T_E)}{(\partial x)_e} \tag{2.12}$$

Considering the control volume surrounding the grid point P is filled with a material of uniform conductivity k_P and around E with a material of conductivity k_E. For this composite slab a steady one dimensional analysis without sources seen in Apendix C leads to:

$$q_e = \frac{T_P - T_E}{\frac{(\partial x)_{e-}}{k_P} + \frac{(\partial x)_{e+}}{k_E}} \tag{2.13}$$

Combinaton of Eq.2.11 and Eq.2.13 leads to a expression of k_e using the factor f_e

$$k_e = \left(\frac{1 - f_e}{k_P} + \frac{f_e}{k_E}\right)^{-1} [1] \tag{2.14}$$

Placing the interface e midway between P and E, we have $f_e = 0.5$ and Eq.2.14 change to

$$k_e^{-1} = 0.5(k_P^{-1} + k_E^{-1}) \tag{2.15a}$$

$$k_e = \frac{2k_P k_E}{k_P + k_E} \tag{2.15b}$$

Equations (2.15) show that k_e is the harmonic mean of k_P and k_E instead the arithmetic mean that Eq. 2.10 wpuld give for a value of $f_e = 0.5$.

2.2.2 Unsteady One-Dimensional Conduction

To make the general formulation for the unsteady One-dimensional conduction we temporally drop the source term for a simpler solving of the govern equation

$$\frac{d}{dx}\left(k\frac{dT}{dx}\right) = \rho c_p \frac{dT}{dt} \tag{2.16}$$

were ρ represents the medium density and c_p the specific heat. For convenience we shall asume ρc_p to be constant.

Since time is a one-way coordinate we obtain the solution by marching in time for a initial distribution of temperature. The problem is to find the values of T at time $t + \Delta t$ for given values of T at time t. We are going to name T_P^0, T_W^0, T_E^0 for "old" values of T at time t and "new" values of T at time $t + \Delta t$ by T_P^0, T_W^0, T_E^0.
We can find the integral resolution of the differential equation in the AppendixB and now we can study the discretization equation (Eq.B.7) for unsteady state without differential terms and getting back the source term dropped:

$$\rho c_p \frac{\Delta x}{\Delta t}(T_P^1 - T_P^0) = f\left[\frac{k_e(T_E^1 - T_P^1)}{(\partial x)_e} + \frac{k_w(T_W^1 - T_P^1)}{(\partial x)_w}\right] + (1-f)\left[\frac{k_e(T_E^0 - T_P^0)}{(\partial x)_e} + \frac{k_w(T_W^0 - T_P^0)}{(\partial x)_w}\right] \quad (2.17)$$

For better understanding and easier formulation we shall drop the superscript 1 and remember that T_P, T_E, T_W stand for the new values of T at the time $t + \Delta t$. Rearranging the terms of Eq.2.17 we can obtain the coefficient formulation:

$$a_P T_P = a_E[fT_E + (1-f)T_E^0] + a_W[fT_W + (1-f)T_W^0] + [a_P^0 - (1-f)a_E - (1-f)a_W]T_P^0 \quad (2.18)$$

where

$$a_E = \frac{k_e}{(\partial x)_e} \quad (2.19a)$$

$$a_W = \frac{k_w}{(\partial x)_w} \quad (2.19b)$$

$$a_P^0 = \frac{\rho c_p \Delta x}{\Delta t} \quad (2.19c)$$

$$a_P = f a_E + f a_W + a_P^0 \quad (2.19d)$$

For certain specific values of the factor f the discretization equation 2.18 reduces to different well known schemes for parabolic differential equations. In the following section we are going to explain the difference between this schemes and two important formulations for discretization equations.

Explicit, Crank-Nicolson and Fully implicit Schemes

In Eq.2.18 we can see that different values of f leads to different schemes and equation forms, $f = 0$ leads to the explicit scheme, $f = 1$ to the fully implicit scheme and $f = 0.5$ to Crank-Nicolson scheme that assumes a linear variation of T_P. For the scope of this work we will only see in detail the fully implicit and explicit scheme.

Explicit scheme essentially assumes that the old value T_P^0 prevails over the entire time step except at time $t + \Delta t$. The fully implicit scheme says that, at time t, T_P suddenly drops from T_P^0 to T_P^1 and remains constant over the whole time step as we can see in Fig.2.4.

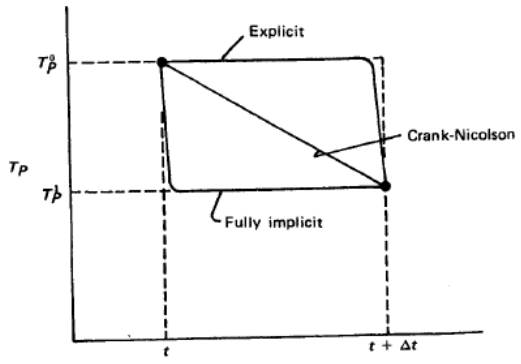


Figure 2.4: Variation of temperature with time for three different schemes[1]

2.3 Convection and Diffusion

Accurate modelling of the interaction between convective and diffusive processes is a challenging task in numerical approximation of partial differential equations. Many different ideas and approaches have been proposed in different contexts in order to resolve the difficulties such as exponential fitting, compact differences, upwind, etc. being some examples from the fields of finite difference and finite element methods.

It is important to know that mathematical models that involve a combination of convective and diffusive processes are among the most widespread in all of science, engineering and other fields where mathematical models are involved. Although convection is the only new term introduced in this section, its formulation is not very simple. The convection term has an inseparable connection with the diffusion term so they need to be handled as one unit. This section gives us a better understanding of the Navier-Stokes equations before treating the final equation that merges the convection and diffusion phenomena. [5][1]

2.3.1 Navier-Stokes Equations

Before starting the formulation of the Convection-Diffusion equation its important to have clear the meaning of each one of the N-S equations. The N-S equations consists of the continuity equation, which represents the mass conservation principle (Eq.2.20); the momentum conservation equations, one for each problem dimension (Eq.2.21); and the energy conservation equation (Eq.2.22). [6]

$$\frac{d\rho}{dt} + \nabla \cdot (\rho \overrightarrow{v}) = 0 \tag{2.20}$$

$$\frac{d}{dt}(\rho\overrightarrow{v}) + \nabla\cdot(\rho\overrightarrow{v}\,\overrightarrow{v}) = -\nabla p + \nabla\cdot(\overrightarrow{\tau}) + \rho\overrightarrow{g} \tag{2.21}$$

$$\frac{d}{dt}(\rho(u+e_c)) + \nabla\cdot((u+e_c)\rho\overrightarrow{v}) = -\nabla\cdot(\rho\overrightarrow{v}) + \nabla\mathring{\mathrm{u}}(\overrightarrow{v}\cdot\overrightarrow{\tau}) - \nabla\cdot\overrightarrow{q} + \rho\overrightarrow{g}\cdot\overrightarrow{v} + G \tag{2.22}$$

Some previous assumptions have been done in the N-S equations, this hypothesis are:

- Continuity of matter
- Continuum medium assumption
- Relativity effects negligible
- Inertial reference system
- Magnetic and electromagnetic forces negligible

Continuity conservation equation

This equation defines that the variation of mass in the control volume has to be equal to the mass flow through its faces. Therefore, taking reference to the terms of Eq. 2.20:

- The first term $\frac{d\rho}{dt}$ represents the variation of mass inside the control volume in a differential time.
- The second term $\nabla\cdot(\rho\overrightarrow{v})$represents the mass flow through the faces of the control volume.

Momentum conservation equation

This equation shows us that the variation of linear momentum in the control volume plus the momentum flux through the CV faces has to be equal to the sum of the forces that act on the CV. Therefore, taking reference to the terms of Eq. 2.21:

- The first term $\frac{d}{dt}(\rho\overrightarrow{v})$ represents the variation of linear momentum in the control volume.
- The second term $\nabla\cdot(\rho\overrightarrow{v}\,\overrightarrow{v})$ represents the momentum flux through the faces of its control volume.
- The third term ∇p is the pressure gradient acting like an axial force on the faces of the CV.
- The fourth term $\nabla\cdot(\overrightarrow{\tau})$ is the total stress tensor. This force acts axially and tangentially on the faces of the control volume. Its value depends on the type of fluid (Newtonian, non-Newtonian...).
- The fifth term $\rho\overrightarrow{g}$ is the volumetric force. This force may be a gravitational, electrical, magnetic or electromagnetic.

Energy conservation equation

This equation defines that the variation of internal energy and kinetic energy in a control volume plus the flow of their variables must be equal to the work done on the control volume plus the incoming heat flow through the faces of the CV plus the energy of the sources in the control volume. Therefore, taking reference to the terms of Eq. 2.22:

- The first term $\frac{d}{dt}(\rho(u+e_c))$ represents the variation of the internal and kinetic energy in the CV.
- The second $\nabla \cdot ((u+e_c)\rho\overrightarrow{v})$ represents the energy flow of these variables through the faces of its volume.
- The third $-\nabla \cdot (\rho\overrightarrow{v})$ and fourth $\nabla \mathring{\mathrm{u}}(\overrightarrow{v} \cdot \overrightarrow{\tau})$ terms are the work done by superficial forces like pressure and stress.
- The fifth $\nabla \cdot \overrightarrow{q}$ term is the incoming heat flow through the faces of the control volume.
- The sixth term $\rho\overrightarrow{g} \cdot \overrightarrow{v}$ represents the work done by the volumetric forces, in this case there is only the gravitational force work.
- The seventh term G is the work done by the internal forces.

2.3.2 Equation Formulation

The Convection-Diffusion equation is a combination of the conservation equations of mass, linear momentum and energy also called Navier-Stokes equations. In the last chapter we did the description of the equation in terms of temperature T and conductivity k now we can easily recast in terms of the general variable ϕ and its diffusion coefficient Γ, the only omission has been the convection term, which we shall now include.

The convection is created by fluid flow, our task is to obtain a solution for ϕ in the presence of a given flow field. Having somehow acquired the flow field we can calculate the temperature, concentration, enthalpy, or any such quantity that is represented by the general variable ϕ.

Simplified N-S Equations

The N-S explained in the previous section can be simplified for the conditions and assumptions related to our Convection-Diffusion equation formulation.[7] Then we can find the simplified N-S that govern the flow of a Newtonian fluid in Cartesian coordinates assuming:

- Two-Dimensional model
- Laminar flow
- Incompressible flow
- Newtonian fluid

- Boussinesq hypothesis[2]
- Negligible viscous dissipation
- Negligible compression or expansion work
- Non-participating medium in radiation
- Mono-component and mono-phase fluid

The use of constant properties of thermal conductivity, density... implies that we will not be able to solve problem with a huge range in temperatures because all of this properties depend on it.

Simplifying equations from Eq.2.20 to 2.22:

$$\frac{du}{dx} + \frac{dv}{dy} = 0 \tag{2.23}$$

$$\rho\frac{du}{dx} + \rho u\frac{du}{dx} + \rho v\frac{du}{dy} = -\frac{dp}{dx} + \mu\Big(\frac{d^2u}{dx^2} + \frac{d^2u}{dy^2}\Big) \tag{2.24}$$

$$\rho\frac{du}{dx} + \rho u\frac{dv}{dx} + \rho v\frac{dv}{dy} = -\frac{dp}{dy} + \mu\Big(\frac{d^2v}{dx^2} + \frac{d^2v}{dy^2}\Big) + \rho g\beta(T - T_\infty) \tag{2.25}$$

$$\rho\frac{dT}{dt} + \rho u\frac{dT}{dx} + \rho v\frac{dT}{dy} = \frac{k}{c_p}\Big(\frac{d^2T}{dx^2} + \frac{d^2T}{dy^2}\Big) + \frac{G}{c_p} \tag{2.26}$$

We can note that in this equation are four unknown values: pressure, temperature and the two components of velocity u and v. Furthermore, a boundary condition and an initial condition are required to solve the problem.

Analyzing the system of equations closely we can notice a strong coupling between them:

- Pressure - Velocity: for the previous established conditions, there is no specific pressure equation, but the pressure distribution allows the velocity field to satisfy the mass conservation equation.
- Temperature-Velocity: there is only a coupling characterization for natural convection, mixed connection or when the physical properties depend on the temperature. In forced convection and constant physical properties, the velocity field does not depend on temperature field.

Convection-Diffusion Equation

Acknowledging the coupling from the partial differential equations and applying the corresponding assumptions all equations from Eq.(2.23 - 2.26) can be summarized in the convection-diffusion equation:

$$\frac{d(\rho\phi)}{dt} + \nabla(\rho\overrightarrow{v}\phi) = \nabla(\Gamma\nabla\phi) + G \tag{2.27}$$

[2]Constant physical properties everywhere except in the body forces term

in Cartesian coordinates, incompressible flow and constant physical properties the equation can also be written as

$$\rho\frac{d\phi}{dt}+\rho u\frac{d\phi}{dx}+\rho v\frac{d\phi}{dy}=\frac{k}{c_p}\Big(\frac{d^2\phi}{dx^2}+\frac{d^2\phi}{dy^2}\Big)+G \tag{2.28}$$

In the previous equation, the first term is the accumulation of ϕ which tells how ϕ change along time. The second and the third term are the net convective flow in the control volume, which gives information about the spatial transport of ϕ. The sum of these has to be equal to the net diffusive flow, which represents the transport of ϕ due to the concentration of gradients, plus the generation of ϕ per unit volume (G). Looking at Eq.2.27 the *diffusion flux* due to the gradient of the general variable ϕ is $-\Gamma(d\phi/dx)$ where ϕ could represent chemical-species diffusion, heat flux, viscous stress, etc.

According to Eq.2.27 we can write a table with the parameters of ϕ, τ and G in order to reproduce the governing equations (Eq. 2.23 - Eq. 2.26).

Equation	ϕ	τ	G
Continuity	1	0	0
Momentum in X direction	u	μ	$-dp/dx$
Momentum in Y direction	v	μ	$-dp/dy+\rho g\beta(T-T_\infty)$
Energy (constant c_p)	T	k/c_p	ϕ/c_p

Table 2.1: Parameters to obtain N-S equations convection-diffusion equation

2.3.3 Equation Discretization

In this section it is shown the implicit finite-volume discretization (FVM) of the convection-diffusion equation. First of all Eq.2.28 has to be integrated into a rectangular CV, Fig.2.5 shows the geometric parameters for the intregration:

$$\frac{(\rho\phi)_P^1-(\rho\phi)_P^0}{\Delta t}\Delta x\Delta y+\Big[(\rho u\phi)_e^1-(\rho u\phi)_w^1\Big]\Delta y+\Big[(\rho v\phi)_n^1-(\rho v\phi)_s^1\Big]\Delta x=$$

$$=\Big[\Big(\Gamma\frac{d\phi}{dx}\Big)_e^1-\Big(\Gamma\frac{d\phi}{dx}\Big)_w^1\Big]\Delta y+\Big[\Big(\Gamma\frac{d\phi}{dy}\Big)_n^1-\Big(\Gamma\frac{d\phi}{dy}\Big)_s^1\Big]\Delta x+G_P^1\Delta x\Delta y \tag{2.29}$$

Note that superindex "1" is used for the value of property ϕ at time $t=t+\Delta t$ and "0" at the previous time step value, for an easier formulation we can state that $\phi^1=\phi$. We assume $\Delta x\Delta y$ as the CV volume V_P and separately as their surfaces S_e , S_w, S_n and S_s

$$\frac{(\rho\phi)_P-(\rho\phi)_P^0}{\Delta t}V_P+\Big[(\rho u\phi)_eS_e-(\rho u\phi)_wS_w\Big]+\Big[(\rho v\phi)_n^1S_n-(\rho v\phi)_sS_s\Big]=$$

$$=\Big[\Big(\Gamma\frac{d\phi}{dx}\Big)_eS_e-\Big(\Gamma\frac{d\phi}{dx}\Big)_wS_w\Big]+\Big[\Big(\Gamma\frac{d\phi}{dy}\Big)_nS_n-\Big(\Gamma\frac{d\phi}{dy}\Big)_sS_s\Big]+G_PV_P \tag{2.30}$$

This formulation can be simplified using the total flux term, defined by:

$$J_x = \rho v \phi - \Gamma \frac{d\phi}{dx} \tag{2.31a}$$

$$J_y = \rho v \phi - \Gamma \frac{d\phi}{dy} \tag{2.31b}$$

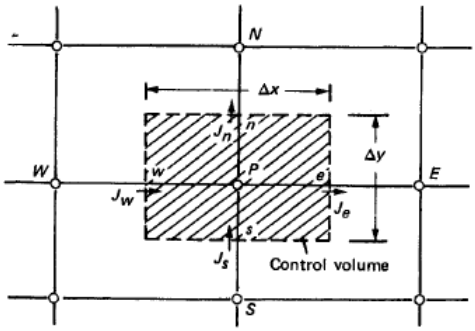


Figure 2.5: Two-Dimensional CV with flux vectors (J)[1]

Equation 2.27 can be expressed with the flux term J as:

$$\frac{d(\rho\phi)}{dt} + \frac{dJ_x}{dx} + \frac{dJ_y}{dy} = G \tag{2.32}$$

Integrating the previous equation into a rectangular CV and assuming an implicit scheme for the temporal integration, Eq. 2.32 yields to:

$$\frac{(\rho\phi)_P - (\rho\phi)_P^0}{\Delta t} V_P + J_e - J_w + J_n - J_s = (G_c + G_P \phi_P) V_P \tag{2.33}$$

The quantities J_e , J_w , J_s and J_n are the integrated total fluxes over the control-volume faces; that is, J_e stands for $\int J_x dy$ over the interface e and so on. The source term has been linearized as we can see in the last term of Eq. 2.33.

We need to note that the flow field has to satisfy the continuity equation (Eq. 2.23) in order to assume convergence:

$$\frac{d}{dx_j}(\rho u_j) = 0 \tag{2.34}$$

Integrating over a rectangular finite volume:

$$\frac{\rho_P - \rho_P^0}{\Delta t} V_P + F_e - F_w + F_n - F_s = 0 \tag{2.35}$$

where F_e, F_n, F_s and F_w are the mass flow rates through the faces of the Control Volume.

$$F_e = (\rho u)_e S_e \tag{2.36a}$$

$$F_w = (\rho u)_w S_w \tag{2.36b}$$

$$F_n = (\rho v)_n S_n \tag{2.36c}$$

$$F_s = (\rho v)_s S_s \tag{2.36d}$$

Multiplying Eq.2.35 by ϕ_P and subtracting it from Eq.2.33:

$$(\phi_P - \phi_P^0)\frac{\rho_P^0}{\Delta t} \cdot V_P + (J_e - F_e\phi_P) - (J_w - F_w\phi_P) + \\ + (J_n - F_n\phi_P) - (J_s - F_s\phi_P) = (S_c + S_P\phi_P) \tag{2.37}$$

The assumption of uniformity over a control-volume face enables us to employ One-Dimensional practices from Ref.[1] for the Two-Dimensional situation.

2.3.4 Numerical Schemes

Numerichal schemes in convection-diffusion problems evaluate the convective and diffusive terms at the CV faces while the dependent variable ϕ is evaluated at the center. Convective flux on any face is given by the arithmetic mean between central node and his neighbours:

$$\left(\frac{d\phi}{dx}\right)_w = \frac{\phi_W - \phi_P}{\delta x_w} \tag{2.38a}$$

$$\left(\frac{d\phi}{dx}\right)_e = \frac{\phi_E - \phi_P}{\delta x_w} \tag{2.38b}$$

$$\left(\frac{d\phi}{dy}\right)_n = \frac{\phi_N - \phi_P}{\delta y_n} \tag{2.38c}$$

$$\left(\frac{d\phi}{dy}\right)_s = \frac{\phi_S - \phi_P}{\delta y_s} \tag{2.38d}$$

Convective and diffusive terms need to be calculated using numerical schemes that evaluates values of ϕ at the nodal points. There are two types of schemes: low order numerical schemes; and high order schemes. The *order* of a numerical scheme is the number of neighbour nodes that are involved to evaluate the dependent variable at the cell face.

For the scope of this project we are only going to treat low order numerical schemes such as: CDS, UDS, HDS, EDS, PLDS. This numerical schemes evaluate the variable using nearest nodes (east (E), west(W), nort(N) and south (S)) with a scheme order of one or two. The *order* of a numerical scheme is defined by the number of neighbouring nodes that are used to evaluate the dependent variable at the cell face. Figure 2.6 shows the values of the variable ϕ given by the different schemes for various values of the Peclet Number (Pe).

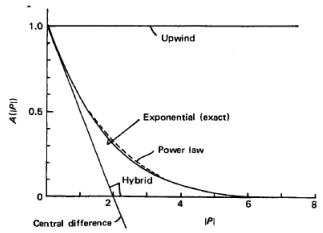


Figure 2.6: The function $A(|Pe|)$ for various low order schemes [1]

Scheme	Formula for $A(\|Pe\|)$
Central difference	$1 - 0.5\|Pe\|$
Upwind	1
Hybrid	$[\![0, 1 - 0.5\|Pe\|\ ]\!]$
Power Law	$[\![0, (1 - 0.1\|Pe\|)^5\ ]\!]$
Exponential	$\|Pe\|/(e^{\|Pe\|} - 1)$

Table 2.2: The function $A(|Pe|)$ for different schemes [1]

Central Difference Scheme (CDS)

It is a second order scheme where the variable at the cell face is calculated as the arithmetic mean of the variable at the neighbour nodes of the face. For the east face of the CV:

$$\phi_e = \frac{1}{2}(\phi_P + \phi_E) \tag{2.39}$$

Using a general notation to reefer to the control volume face or center:

$$\phi_{if} = \frac{1}{2}(\phi_P + \phi_{ib}) \tag{2.40}$$

Looking at Fig.2.6 we note that all schemes except the CDS give physically realistic solutions because it can produce values that lie outside the $[0 - 1]$ range established by the Scarborough criterion, see Appendix E. We can find the formula of $A(|Pe|)$ for this scheme in Table 2.2:

$$A(|Pe_{if}|) = 1 - 0.5|Pe_{if}| \tag{2.41}$$

Upwind Difference Scheme (UDS)

It is a first order scheme where the value of ϕ at the cell face is equal to the value of ϕ at the grid point on the upwind side of the face.

$$\phi_e = \phi_P \quad if \quad F_e > 0 \tag{2.42a}$$

$$\phi_e = \phi_E \quad if \quad F_e < 0 \tag{2.42b}$$

What that means is that if u is positive, the value of ϕ at the face will be the value of ϕ at the left grid point. However, if u is negative, the value of ϕ at the face cell will be the value of ϕ at the right grid point. It will be the same reasoning for v. It is defined a new operator for this criterion as $[\![A, B]\!]$. Thus,

$$F_e\phi_e = \phi_P[\![F_e, 0]\!] - \phi_E[\![-F_e, 0]\!] \tag{2.43}$$

This scheme solves the problem that the CDS has because all the coefficients in the equations are always positive or null. That means that the Scarborough criterion is always satisfied. We can find the formula of $A(|Pe|)$ for this scheme in Table 2.2:

$$A(|Pe_{if}|) = 1 \tag{2.44}$$

Hybrid Difference Scheme (HDS)

It is a combination of central difference scheme and upwind difference scheme as it exploits the favorable properties of both of these schemes. This scheme uses CDS for low velocities and UDS for high velocities, it consists in approximating the value of the dimensionless form of a_E (Eq. 2.45) to three linear zones. Fig. 2.7 shows this approximation.

$$\frac{a_E}{D_e} = \frac{Pe_e}{exp(Pe_e) - 1} \tag{2.45}$$

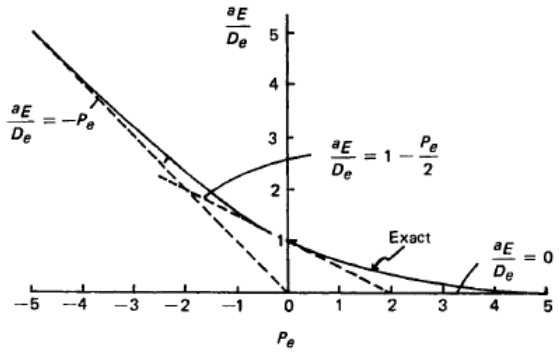


Figure 2.7: Variation of coefficient a_E with Peclet number [1]

For positives values of Pe_e the grid point E is the *downstream* neighbor and its influence is seen to decrease as Pe_e increases. When Pe_e is negative the point E is the *usptream* neighbor and has a large influence. The tree straight lines represent the three limiting cases, they can be seen to form an envelope of, and represent a reasonable approximation to, the exact curve. Then,

For $Pe_e < -2$,

$$\frac{a_E}{D_e} = -Pe_e \tag{2.46}$$

For $-2 \leq Pe_e \leq -2$,

$$\frac{a_E}{D_e} = 1 - \frac{Pe_e}{2} \tag{2.47}$$

For $Pe_e > 2$,

$$\frac{a_E}{D_e} = 0 \tag{2.48}$$

This expressions can be compacted into the following form[3]:

$$a_E = D_e \Big[\!\Big[-Pe_e, 1 - \frac{Pe_e}{2}, 0 \Big]\!\Big] \tag{2.49a}$$

$$a_E = \Big[\!\Big[-F_e, D_e - \frac{F_e}{2}, 0 \Big]\!\Big] \tag{2.49b}$$

We need to note that it is identical with the central-difference scheme for the Peclet number range $-2 \leq Pe_e \leq 2$, and outside this range it reduces to the upwind scheme in which the diffusion has been set equal to zero. For that reason and from Table 2.2 the function of $A(|Pe|)$ is

$$A(|Pe_i f|) = [\![0, 1 - 0.5|Pe_i f|]\!] \tag{2.50}$$

Exponential Difference Scheme (EDS)

It is a second order scheme where the evaluation of the variables at the cell faces come from te exact solution of the Eq. 2.28 for the steady One-Dimensional problem without source term [7]. From [1] we know that exact solution is:

$$\frac{\phi - \phi_0}{\phi_L - \phi_0} = \frac{exp(Pe \cdot x/L) - 1}{exp(Pe) - 1} \tag{2.51}$$

Where ϕ_0 is the value of ϕ at the left boundary ($x = 0$); ϕ_L the value of ϕ at the right boundary ($x = L$); x is the position of the left interface node; Pe is the Peclet number; and L is the distance of the domain ($0 \leq x \leq L$). Remember that the Peclet number is defined by:

$$Pe \equiv \frac{\rho u L}{\Gamma} \tag{2.52}$$

From Eq.2.51 it can be seen that P is the ratio of strengths of convection and diffusion, which gives us a better understanding about the meaning of this number inside the case of study. The nature of Eq.2.51 ac be understood from Fig. 2.8 where the variation of $\phi \sim x$ for different values of the Peclet number is shown.

[3]This special symbol [[]]stands for the largest of the quantities contained within it.

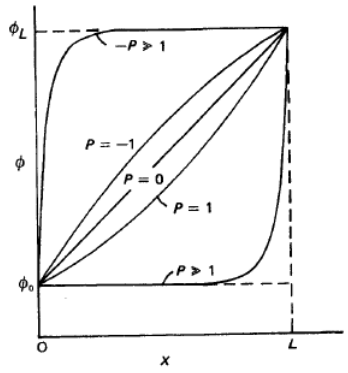


Figure 2.8: Exact solution for the one-dimensional convection-diffusion problem [1]

This scheme gives an exact solution for 1D for any Peclet number, although it is not exact for the 2D and 3D situations. Another disadvantage would be the extra time it takes to compute the solution with exponential functions. We can find the formula of $A(|Pe|)$ for this scheme in Table2.2:

$$A(|Pe_{if}|) = |Pe_i f|/(e^{|Pe_i f|} - 1) \tag{2.53}$$

Power-law Difference Scheme (PLDS)

Taking the HDS it seems a little premature to set the diffusion effects equal to zero as soon as the Peclet number exceeds 2, a better aproximation to the exact curve is givven by the power-law scheme. It is a second order scheme where the variable at the cell face is calculated with an approximation of the EDS by a polinomial of fifth degree.

From [1] we can get the compact form for the coefficient a_E:

$$a_E = D_e \left[\!\left[0, \left(1 - \frac{0.1|F_e|}{D_e}\right)^5\right]\!\right] + [\![0, -F_e]\!] \tag{2.54}$$

We can find the formula of $A(|Pe|)$ for this scheme in Table2.2:

$$A(|Pe_{if}|) = [\![0, (1 - 0.1|Pe_i f|)^5]\!] \tag{2.55}$$

2.3.5 Final Discretization Equation

From Eq. 2.37 and according to [1, 7] the Two-Dimensional final discretization equation can now be rewritten as

$$a_P \phi_P = a_E \phi_E + a_S \phi_S + a_W \phi_W + a_N \phi_N + b \tag{2.56}$$

Where the coefficients a_i can be evaluated as:

$$a_E = D_e \cdot A(|Pe_e|) + [\![-F_e, 0]\!] \tag{2.57a}$$
$$a_W = D_w \cdot A(|Pe_w|) + [\![F_w, 0]\!] \tag{2.57b}$$
$$a_N = D_n \cdot A(|Pe_n|) + [\![-F_n, 0]\!] \tag{2.57c}$$
$$a_S = D_s \cdot A(|Pe_s|) + [\![F_s, 0]\!] \tag{2.57d}$$
$$a_P^0 = \frac{\phi_P^0 V_P}{\Delta t} \tag{2.57e}$$
$$b = G_C V_P + a_P^0 \phi_P^0 \tag{2.57f}$$
$$a_P = a_E + a_S + a_W + a_N + a_P^0 - G_P V_P \tag{2.57g}$$

Where the flow rates through the faces (F_{if}) are:

$$F_e = (\rho u)_e S_e \tag{2.58a}$$
$$F_w = (\rho u)_w S_w \tag{2.58b}$$
$$F_n = (\rho v)_n S_n \tag{2.58c}$$
$$F_s = (\rho v)_s S_s \tag{2.58d}$$

The corresponding conductances are defined by

$$D_e = \frac{\Gamma_e S_e}{(\delta x)_e} \tag{2.59a}$$
$$D_w = \frac{\Gamma_w S_w}{(\delta x)_w} \tag{2.59b}$$
$$D_n = \frac{\Gamma_n S_n}{(\delta x)_n} \tag{2.59c}$$
$$D_s = \frac{\Gamma_s S_s}{(\delta x)_s} \tag{2.59d}$$

and the Peclet numbers by

$$P_e = \frac{F_e}{D_e} \quad P_n = \frac{F_n}{D_n} \quad P_s = \frac{F_s}{D_s} \quad P_w = \frac{F_w}{D_w} \tag{2.60}$$

We need to note that all the formulation has been done in order that the value of $A(|Pe_i f|)$ depends on the numerical scheme used. This value can be found in Table 2.2.

2.4 Boundary Conditions

Still need to be filled Dirichet i Newman

2.5 Main Algorithm

Still need to be filled
node si posarte aqui seccio tete joder, si no vas aqui vas al final del seguent capitul

Chapter 3

Numerical Analysis

Contents

Some of the main ingredients[8] of a *numerical solution method* have been introduced in the previous section as we introduce the physical phenomenons studied. In this section they are presented and some developed in order to give us a better comprehension about the implementation of the previously developed concepts. It is needed in order to have a clear idea in what this methods consists before developing the simulations for each case of study.

3.1 Mathematical Model

The starting point of any numerical method is the mathematical model. As it has been seen in Chapter 2 this modeling led us to a set of partial differential equations needed for the aim of the study. Choosing a model also consists in making the correct assumptions such as the domain dimensions and the nature of the fluid of study. As already seen in Section 2.3.1 our model includes simplifications of the conservation laws, leading to a simplified model. The assumptions took into account for developing the models for the cases of study were:

- Continuity of matter
- Continuum medium assumption
- Relativity effects negligible
- Inertial reference system
- Magnetic and electromagnetic forces negligible
- Laminar flow
- Incompressible flow
- Constant conductivity k
- Newtonian fluid
- Boussinesq hypothesis
- Negligible viscous dissipation
- Negligible compression or expansion work
- Non-participating medium in radiation
- Mono-component and mono-phase fluid

In the mathematical model and approach to the phenomena of the study it has been developed a general formulation for getting a general solution method applicable to different problems. This general formulation is simplified for each case of study, as with most general purpose tools, they are usually not optimum for any one application. [8]

3.2 Discretization Method

After selecting the mathematical model, it is needed to chose a suitable discretization method for approximating the differential equations by a system of algebraic equations for the variables at some discrete locations in the space and time. The most important discretization methods are: finite difference (FDM), finite volume (FVM) and finite element (FEM) methods. Each type of method yields the same solution if the grid is fine enough. However, some methods are more suitable to some classes problems than others. For the aim of this thesis we are only going to develop the two first models aforementioned. For more information see [8].

3.2.1 Finite Difference Method

This is the oldest and also the simplest discretization method for numerical solutions of PDE's sets. The starting point is the conservation equation in differential form. As it is shown in Section 3.4 the domain is covered by a grid. The usual procedure for deriving finite-difference equations[1] consists of approximating tge derivatives in the differential equation using the Taylor-series expansion. Considering the grid from Fig.3.1, for the grid point 2 the expansion around it would be:

$$\phi_1 = \phi_2 - \Delta x\Big(\frac{d\phi}{dx}\Big)_2 + \frac{1}{2}(\Delta x)^2\Big(\frac{d^2\phi}{dx^2}\Big)_2 - \dots \tag{3.1}$$

$$\phi_3 = \phi_2 + \Delta x\Big(\frac{d\phi}{dx}\Big)_2 + \frac{1}{2}(\Delta x)^2\Big(\frac{d^2\phi}{dx^2}\Big)_2 + \dots \tag{3.2}$$

Cuting the series after the third term and after working this equations, we get

$$\Big(\frac{d\phi}{dx}\Big)_2 = \frac{\phi_3 - \phi_1}{2\Delta x} \tag{3.3}$$

$$\Big(\frac{d\phi}{dx}\Big)_2 = \frac{\phi_1 + \phi_3 - 2\phi_2}{(\Delta x)^2} \tag{3.4}$$

The substitution of this expressions into the govern differential equations of each phenomena leads to the finite-difference equation. This method assumes in some way that the higher derivatives of ϕ are unimportant, which could lead to an undesirable formulation in some cases. On structured grids, the FDM method is very simple and effective. The disadvantages of this method is that the conservation is not enforced unless special care is taken and the restriction to simple geometries. See [9, 8] for better comprehension.

3.2.2 Finite Volume Method

3.3 Coordinate and Vector System

3.4 Numerical Grid

No specific information has been provided as to where the control volume faces are located in relation to the grid points, since the discretization equations have been displayed in general terms so that it will be applicable to any particular way of locating the control volume faces. There are

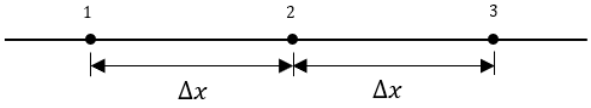


Figure 3.1: Grid points for the Taylor series expansion

many possible ways for locating the control-volume but for the aim of this thesis we are going to focus in two different grids. The description of each one will refer to a two-dimensional situation, although the concepts involved are applicable to one and three-dimensional situations.

3.4.1 Grid A: Faces located midway between the grid points

One of the most intuitive practise to construct the control volume is to place their faces *midway* between neighboring grid points as we can see in Fig.3.2a. For building this grid to a 2-D plate we should place grid points on his boundaries. Another observation is that the grid is nonuniform; on consequence the grid point P does not lie at the geometric center of the control volume.

3.4.2 Grid B: Grid points placed at the centers of the control-volumes

Another way to draw the grid is to draw the control-volume boundaries first and then place the grid point at the geometric center of each control-volume. As we can see in the Fig.3.2b when the control volume sizes are non-uniform, their faces does not lie midway between the grid points.

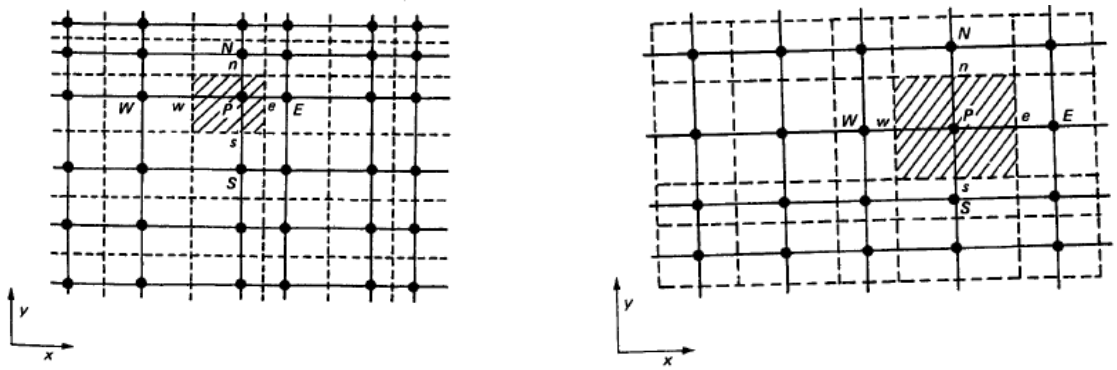


Figure 3.2: Location of control-volume faces (a)Grid A (b)Grid B [1]

The fact that the grid point P in Fig.3.2a may not be at the geometric center of the control volume represents a disadvantage. That is because the temperature T_P cannot be observed as good representative value for the control volume in the calculation of the source term, the conductivity, and similar quantities[1]. The Grid A also presents objections in the calculation of the heat fluxes at the control volume faces, if we take grip point e in Fig.3.2a, for example, we can se that it is not at the center of the control-volume face i which it lies. Then, we assume that the heat flux at e prevails over the entire face brings some inaccuracy.

Grid B does not have this problems because the point P lies at the center of the control volume by definition and points such as e lies at the center of their respective faces. One of the decisive advantages of Grid B is that the control volume turns out to be the basic unit of the discretization method, it is more convenient drawing the control-volume boundaries first and let the grid-point

locations follow as consequence.

There are some advantages from the Grid A over Grid B but the aforementioned advantages that Grid B represents over the grid A makes us consider that the election of Grid B for our problem formulation is going to be the most suitable. We need to make additional considerations for the control volume near the boundaries of the domain. In the chosen case (Grid B) it is convenient to completely fill the calculation domain with regular control volumes and to place the boundary grid points on the faces of the near-boundary control volume faces. We can see this arrangement of the Grid B in Fig. 3.3 where a typical boundary face i is located not between the boundary point B and the internal point I, it actually passes through the boundary point.

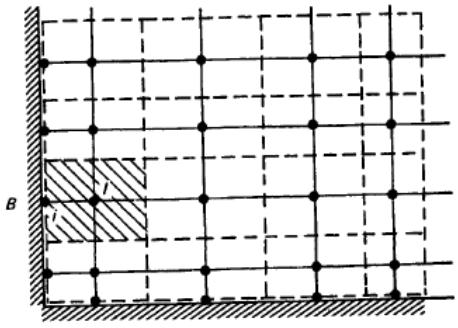


Figure 3.3: Boundary control volumes in Practice B[1]

3.5 Finnite Approximations

3.6 Numerical Solver Methods

3.7 Numerical Solver Methods

In this section we will see some of the methods studied for the aim of this project. Each method has their advantages, disadvantages and applications that depends directly to the problem approach. Different methods applied to the same problem would give us similar results if they are correctly used, the difference between them lies in the computational cost. There are situations where you have a limited time to run your code and you need to be able to get all the needed data for the study, in this case a faster solver would be required.

At this stage it is known that our solver will need to solve a system of algebraic equations in order to obtain the field of the parameter ϕ until convergence in the solutions is achieved.

$$A \cdot x = b \tag{3.5}$$

where A is the matrix with the dependent coefficients, x is a vector that contains the parameter of study and b is the non-dependent coefficient.

If we had to solve this system in the mathematical way we would need to do the inversion of the matrix A, this procedure would suppose a too high computational cost making it unfeasible. This is why there are other ways to solve this system of equations and why each solver have their own characteristics such as convergence velocity, simplicity, possibility to be paralleled, etc.

Before explaining in what each solver consists its important to know how the matrix A is formed and how it is structured. For this we are going to use an example 3 by 3 grid, the smallest one for treating all the possible cases, we can see it in Fig. 3.4.

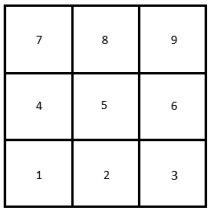


Figure 3.4: 3 by 3 example grid

Each row of the matrix A contains their corresponding coefficients, first row would contain first grid point coefficients and so on. This means that most of the matrix A parameters are equal to 0. The grid point or node coefficients are: a_P, a_N, a_S, a_E and a_W.

$$A = \begin{Bmatrix} a_{P1} & a_{E1} & 0 & a_{N1} & 0 & 0 & 0 & 0 & 0 \\ a_{W2} & a_{P2} & a_{E2} & 0 & a_{N2} & 0 & 0 & 0 & 0 \\ 0 & a_{W3} & a_{P3} & 0 & 0 & a_{N3} & 0 & 0 & 0 \\ a_{S4} & 0 & 0 & a_{P4} & a_{E4} & 0 & a_{N4} & 0 & 0 \\ 0 & a_{S5} & 0 & a_{W5} & a_{P5} & a_{E5} & 0 & a_{N5} & 0 \\ 0 & 0 & a_{S6} & 0 & a_{W6} & a_{P6} & 0 & 0 & a_{N6} \\ 0 & 0 & 0 & a_{S7} & 0 & 0 & a_{P7} & a_{E7} & 0 \\ 0 & 0 & 0 & 0 & a_{S8} & 0 & a_{W8} & a_{P8} & a_{E8} \\ 0 & 0 & 0 & 0 & 0 & a_{S9} & 0 & a_{W9} & a_{P9} \end{Bmatrix} \tag{3.6}$$

Working with this matrix form allows you to understand why each term is located in their corresponding position, for noticing that you need to take care of each node neighbour and the A matrix multiplication by the parameter vector x which gives us the govern equation for each point. As we can see in the row number five appear all the coefficients because at this grid point you are surrounded by all the possible neighbor points.

Once understood the system of equations and the way we construct the coefficient matrix we can start explaining the different types of solvers available for solving this sets of algebraic equations. As it is seen in the following subsection, there are many solvers that can be classified into two big groups, the direct solvers and the iterative solvers.

3.7.1 Direct Solvers

This group of solvers works all the equations directly finding the solution of the problem at the first iteration over the grid points. In general this solvers are complex to program and require more memory than the iterative solvers because all the domain coefficients need to be saved in order to compute the final result, although they are fast algorithms often used.

Lower-Upper Decomposition (LU)

In numerical analysis and linear algebra, lower-upper (LU) decomposition or factorization factors a matrix as a product of a lower triangular matrix and an upper triangular matrix using the multipliers from Gaussian elimination[1], such that the product of this two matrices gives the original matrix:

$$A = L \cdot U \tag{3.7}$$

Where L is the lower triangular matrix with diagonal elements being equal to 1 and U is the upper diagonal matrix formed as a result of applying Gauss Elimination Method. For the following deductions see [11, 12]. Introducing Eq.3.7 into Eq. 3.5:

$$L \cdot U \cdot x = b \tag{3.8}$$

taking

$$Z = U \cdot x \tag{3.9}$$

where Z is the matrix of artificial variables. Finally we only need to solve two matrix equations in order to break the transformation and obtain the values of the parameter x.

$$L \cdot Z = b \tag{3.10}$$

$$U \cdot x = Z \tag{3.11}$$

Tri-Diagonal Matrix Algorithm (TDMA)

The Tri-Diagonal Matrix Algorithm, also known as the Thomas algorithm, is a simplified form of Gaussian eliminations that can be used for solving tri-diagonal systems of equations. In this systems, all the nonzero coefficients align themselves along three diagonals of the matrix:

$$a_P x_i + a_E x_{i+1} + a_W x_{i-1} = b_i \qquad i = 1 \div n \tag{3.12}$$

This algorithm by itself can be only applyed to 1D situations. It consists in decomposing the matrix A as a product of two triangular matrices L and U following the same process as the LU solver. In the forward-substitution process we seek a relation

$$x_i = P_i x_{i+1} + Q_i \tag{3.13}$$

having obtained from it this relation

$$x_{i-1} = P_{i-1} x_i + Q_{i-1} \tag{3.14}$$

[1]For more information about Gaussian Elimination see [10]

Substitution of Eq.3.14 into Eq.3.12 leads to

$$a_{Pi}ix_i = a_{Ei}x_{i+1} + a_{Wi}(P_{i-1}x_i + Q_{i-1}) + a_{Wi} \tag{3.15}$$

which can be rearranged to look like Eq.3.13. In other words, the coefficients P_i and Q_i then stand for

$$P_i = \frac{a_{Ei}}{a_{Pi} - c_i P_{i-1}} \tag{3.16a}$$

$$Q_i = \frac{b_i + a_{Wi}Q_{i-1}}{a_{Pi} - a_{Wi}P_{i-1}} \tag{3.16b}$$

After this process now we are in a position to start the back substitution via Eq.3.13, for more information about the developement of this method see [1]. Once the formulation is done it can be implemented into an algorithm.

Algorithm

1. Set P_1 and Q_1
2. Use the recurrence relations (3.16) to obtain P_i and Q_i for $i = 2, 3, ..., n$
3. Set $x_{n+1} = Q_{n+1}$
4. Use Eq.3.13 for $i = n, n-1, n-2, ..., 3, 2$ to obtain x_i at this grid points.

Note that this algorithm is applied to the grid-point cluster chosen for our cases of study. The tri-diagonal algorithm is a very powerful and convenient equation solver whenever the equations can be represented in the coefficient form of Eq.3.12. Unlike general matrix methods, the TDMA requires computer storage and computer time proportional only to N, rather than N^2 or N^3.

3.7.2 Iterative Solvers

The set of linear equations obtained from the govern equations of the phenomenon that we are modeling is most often solved iteratively rather than using Gaussian Elimination. In an iterative solver the unknown parameter x_i is not determined simultaneously for all the domain. Instead, some unknowns are determined using guessed values and using this values it can be computed the parameter x_i repeating this process till the convergence of the results.[13]

In other words, it attempts to solve the system of equations by successive approximations to the solution starting from an initial estimated value. The process converges when an approximate solution is found after a huge number of iterations. In general this methods are useful for solving big domains because they don't require a big memory space for saving the matrix coefficients. In the following lines we can find some examples of iterative solvers.

Jacobi's Method

It perhaps is the simplest iterative method for solving $A \cdot x = b$. Note that the simplicity of the method is both good and bad: good because it is relatively easy to understand and thus is a good first taste of iterative methods; bad because it is not typically used in practice although there are potential applications to be considered in the field of parallel computing. The strategy of Jacobi's Method shown in [2] is to use the first equation and the current values of x_i

$$x_i^{(k)} = (x_1^{(k)} x_2^{(k)}, x_3^{(k)}, x_4^{(k)}, ..., x_n^{(k)}) \tag{3.17}$$

to find a new value $x_1^{(k+1)}$ and similarly find a new value $x_i^{(k)}$ using the i^{th} equation and the old values of the other variables. To be clear, the subscript i means that $x_i^{(k)}$ is the i^{th} element of the vector x and superscript k corresponds to the particular iteration.

Given the current values $x_i^{(k)} = (x_1^{(k)} x_2^{(k)}, x_3^{(k)}, x_4^{(k)}, ..., x_n^{(k)})$ we need to find the new values

$$x_i^{(k+1)} = (x_1^{(k+1)} x_2^{(k+1)}, x_3^{(k+1)}, x_4^{(k+1)}, ..., x_n^{(k+1)}) \tag{3.18}$$

in the following system of eqations

$$\begin{array}{ccccccccc} a_{11}x_1^{k+1} & + & a_{12}x_2^{k} & + & \dots & + & a_{1n}x_n^{k} & = & b_n \\ a_{21}x_1^{k} & + & a_{22}x_2^{k+1} & + & \dots & + & a_{2n}x_n^{k} & = & b_2 \\ \vdots & & \vdots & & \ddots & & \vdots & & \vdots \\ a_{n1}x_1^{k} & + & a_{n2}x_2^{k} & + & \dots & + & a_{nn}x_n^{k+1} & = & b_n \end{array} \tag{3.19}$$

which has the same form as Eq.3.5 but a different nomenclature for a better understanding. This system can also be written in his matrix form and decompose matrix A into matrices D, L and U as we can see in Fig.3.5. Where D is a diagonal matrix, L strict lower triangular and U strict upper triangular.

$$\begin{bmatrix} a_{11} & 0 & \cdots & 0 \\ 0 & a_{22} & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & a_{nn} \end{bmatrix} \begin{bmatrix} x_1^{(k+1)} \\ x_2^{(k+1)} \\ \vdots \\ x_n^{(k+1)} \end{bmatrix} + \begin{bmatrix} 0 & a_{12} & \cdots & a_{1n} \\ a_{21} & 0 & \ddots & \vdots \\ \vdots & \ddots & \ddots & a_{n-1\,n} \\ a_{n1} & \cdots & a_{n\,n-1} & 0 \end{bmatrix} \begin{bmatrix} x_1^{(k)} \\ x_2^{(k)} \\ \vdots \\ x_n^{(k)} \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix}.$$

$$D = \begin{bmatrix} a_{11} & 0 & \cdots & 0 \\ 0 & a_{22} & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & a_{nn} \end{bmatrix}, \; L = \begin{bmatrix} 0 & 0 & \cdots & 0 \\ a_{21} & 0 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ a_{n1} & \cdots & a_{nn-1} & 0 \end{bmatrix} \text{ and } U = \begin{bmatrix} 0 & a_{12} & \cdots & a_{1n} \\ 0 & 0 & \ddots & \vdots \\ \vdots & \ddots & \ddots & a_{n-1\,n} \\ 0 & \cdots & 0 & 0 \end{bmatrix},$$

Figure 3.5: Matrix form for Eq.3.19 and D, U and L matrices[2]

Finally, Jacobi's Method can be written in his matrix-vector notation as

$$D \cdot x_i^{(k+1)} + (L + U) x_i^{(k)} = b_i \tag{3.20}$$

so that

$$x_i^{(k+1)} = D^{-1} \cdot [(-L - U)x^{(k)} + b] \tag{3.21}$$

Once the formulation is done it can be implemented into an algorithm.

Algorithm

1. Find $x_1^{(k+1)}$ for the given values of $x_i^{(k)}$.
2. Do the same for $x_i^{(k+1)}$
3. Repeat till convergence

Gauss-Seidel

Gaus-Seidel Method, also called point-by-point method, consists in taking the Jacobi's Method one step further. It's known that $x_1^{(k+1)}$ is a better approximation to the true value of x_1 than $x_1^{(k)}$ is, then it would make sense that once we have found the *new* value $x_1^{(k+1)}$ to use it rather than the old value in finding $x_2^{(k+1)}$.[14]

So $x_1^{(k+1)}$ is found as in Jacobi's Method, but in finding $x_2^{(k+1)}$ we instead use the *new* value $x_1^{(k+1)}$ and the old values

$$x_i^{(k)} = x_3^{(k)}, x_4^{(k)}, ... \ , x_n^{(k)} \tag{3.22}$$

Same as in Jacobi's Method, the linear system decomposes in the same matrices and matrix shape (Eq.3.20-3.21). It is important to note that this method not always converge but applying the Scarborough Criterion (Appendix E) guarantees the convergence of our system of equations. It also has the disadvantage due to the fact that the boundary information is transmitted at the rate of one grid position for each iteration of the solver what gives us a slow convergence.

Example

Taking the implicit 2D heat conduction equation (Eq.4.4) from Section 4.1.3:

$$T_P a_P = T_N a_N + T_S a_S + T_W a_W + T_E a_E + b \tag{3.23}$$

We can rewrite the equation as

$$a_P T_P = \Sigma a_{ib} T_{ib} + b \tag{3.24}$$

Remember that the subscript ib denotes a neighbor point and T_P is the parameter of x_i for the visited grid point, calculated from

$$T_P = \frac{\Sigma a_{ib} T_{ib}^{st} + b}{a_P} \tag{3.25}$$

Where T_{ib}^{st} is the neighbor-point value present in the computer storage, the same as $x_i^{(k+1)}$. This example clarifies the application of the iterative methods explained in this section to a a practical

case. Once the formulation is done it can be implemented into an algorithm.

Algorithm

1. Solve the equations of Eq. System 3.19 for each grid point with the *new* values of $x_i^{(k+1)}$ for their corresponding equation.
2. Iterate over Eq. System 3.19 with the *new* values obtained from the previous iteration until the convergence of the results.
3. Check convergence of all the grid points $|x_i^{(k+1)} - x_i^{(k)}| < \delta$

Line-by-Line Method

This method is a convenient combination of the TDMA direct method for One-Dimensional situations and the Gauss-Seidel Method. It is used when the matrix A is composed by more than three non-null diagonals, in this case the TDMA can not work alone.[1]

Focusing in the Two-Dimensional situation, there are 5 or more no-null diagonals depending on the numerical scheme, as we can see in Eq. 3.6. For solving this system of equations we need to convert a 2D case into a 1D case solved by a TDMA algorithm. For that we shall choose a grid line (Fig.3.6) and assume that the parameter of study x_i along the neighboring lines is known from his *new* value and solve for the x_i's along the chosen line by the TDMA. This procedure needs to be followed for all the lines in one direction and repeated if desired for the lines in the other direction(s). Although we are referring to the 2D situation this method is equally applicable to 3D case.

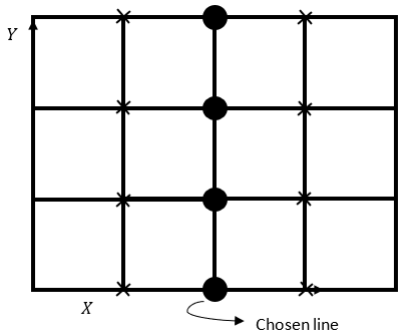


Figure 3.6: Representation of the line-by-line method

The convergence of the line-by-line method is faster than the Gauss-Seidel Method because the boundary-condition information from the ends of the line is transmitted at once to the interior of the domain, no matter how many grid points lie along the line. Although the rate of transmission of information in the other direction is similar to that of the point-by-point method. By alternating the directions in which the TDMA traverse is employed we can quickly bring the information from all the boundaries to the interior. For more information see [1].

Example

Taking the implicit 2D heat conduction equation (Eq.4.4) from Section 4.1.3 it could be rewrote as

$$T_P a_P + T_N a_N + T_S a_S + T_W a_W + T_E a_E = \overline{b} \tag{3.26}$$

then applying the line-by-line method criteria we can write the previous equation into his final form:

$$T_P a_P + T_N a_N + T_S a_S = b \tag{3.27}$$

Where:

$$b = \overline{b} - T_W a_W - T_E a_E \tag{3.28}$$

We can use the same procedure for the x-direction line-by-line application. Once the formulation is done it can be implemented into an algorithm.

Algorithm

1. Solve vertically the line-by-line algorithm for $j = 1, 2, ..., n$
2. Save the found value of $x_i^{(k)}$ into the computer storage $x_i^{(k+1)} = x_i^{(k)}$
3. Solve horizontally the line-by-line algorithm for $i = 1, 2, ..., n$
4. Check convergence of all the grid points $|x_i^{(k+1)} - x_i^{(k)}| < \delta$.
   (a) If not achieved save the found value of $x_i^{(k)}$ into the computer storage $x_i^{(k+1)} = x_i^{(k)}$ and go to step 1.
   (b) If achieved stop the solver and continue the code.

If we do not care about the convergence speed we can skip the third step. It is seen that step 1 and 3 could be switched in order to improve the convergence speed, this depends on the domain dimensions (See [1] for more information).

For concluding this section its important to note that direct solvers are more accurate and faster than the iterative ones. On the other hand iterative solvers are easier to implement in the program than direct solvers for its simplicity. For the scope of this study only iterative solvers are going to be implemented in the code.

3.8 Convergence Criteria

Chapter 4

Cases of Study

Contents

In this chapter we are going to explain each case of study selected for this Grade Final Thesis explaining the objective, problem definition, discretization, algorithm and results.
Still need to be filled

4.1 Two-Dimensional Unsteady Heat Transfer Conduction

The objective of this section is the application of the heat conduction formulation stated in Section 2.2 to a two-dimensional heat transfer conduction problem proposed by the CTTC.

4.1.1 Problem Definition

It is required to solve a two-dimensional transient heat conduction problem which consists in a very long rod composed of four different materials (M1 to M4) for the boundary conditions given in Table 4.3. For each case a plot with the temperature field of the rod will be presented in order to check the validity of the results and the code. The Fig. 4.1 shows the schema of the proposed problem and we can find the properties for each material in Table 4.2. The initial temperature field has to be set at $T = 8.00°C$.

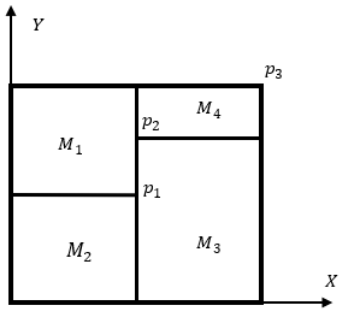


Figure 4.1: General schema of the proposed problem

We can find the needed coordinates of the domain in the following table

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

Table 4.1: Domain coordinates for the first case of study

	$\rho[kg/m^3]$	$c_p[J/kgK]$	$k[W/mK]$
M_1	1500.00	750.00	170.00
M_2	1600.00	770.00	140.00
M_3	1900.00	810.00	200.00
M_4	2500.00	930.00	140.00

Table 4.2: Material properties for the first case of study

4.1.2 Boundary Conditions

The Table 4.3 shows the boundary conditions for the final results of this case of study. Before getting the correct results for the study other boundary conditions and domain distribution is applyed in order to veryfy the code.

Boundary	Boundary condition
Bottom	Isoterm at $T = 23.00°C$
Top	Uniform $Q_f low = 60.00W/m$ length
Left	In contact with a fluid $T_g = 33.00°C$ with $h = 9.00W/m^2K$
Right	Uniform temperature $T = 8.00 + 0.005t°C$

Table 4.3: Boundary conditions for the first case of study

4.1.3 Discretization

From the general formulation (Eq.2.16) for the unsteady one-dimensional conduction we can write the govern equation for this case

$$\frac{d}{dx}\Big(k\frac{dT}{dx}\Big)\frac{d}{dy}\Big(k\frac{dT}{dy}\Big) + \overline{G} = \rho c_p \frac{dT}{dt} \tag{4.1}$$

As we have said in Section 2.2.2 we are going to focus in the analysis of the fully implicit and explicit scheme. Each scheme leads to different forms of the discretization equation, from Eq.2.17 adding the extra dimension and the source term we can rebuild the coefficient equation to a new formulation. We can also do an energy balance developed in Appendix D to obtain Eq.4.2 for a better comprehension of the phenomenon.

$$a_W T_W + a_E T_E + a_N T_N + a_S T_S - T_P(a_W + a_E + a_N + a_S) + G\Delta x\Delta y = \rho c_p \frac{\Delta x\Delta y}{\Delta t}(T_P^1 - T_P^0) \tag{4.2}$$

With coefficients

$$a_W = \frac{k_w \Delta y}{(\partial x)_w} \tag{4.3a}$$

$$a_E = \frac{k_e \Delta y}{(\partial x)_e} \tag{4.3b}$$

$$a_N = \frac{k_n \Delta x}{(\partial y)_n} \tag{4.3c}$$

$$a_S = \frac{k_s \Delta x}{(\partial y)_s} \tag{4.3d}$$

Fully Implicit Formulation

From Eq.4.2 taking T_P^1 as T_P and reordering terms we get

$$T_P a_P = T_N a_N + T_S a_S + T_W a_W + T_E a_E + b \tag{4.4}$$

where

$$a_P = a_W + a_E + a_N + a_S + a_P^0 - G_P \Delta x \Delta y \tag{4.5a}$$

$$a_P^0 = \rho c_p \frac{\Delta x \Delta y}{\Delta t} \tag{4.5b}$$

$$b = G_C \Delta x \Delta y + a_P^0 T_P^0 \tag{4.5c}$$

Explicit Formulation

From Eq.4.2 reordering terms we get

$$T_P a_P = T_N^0 a_N + T_S^0 a_S + T_W^0 a_W + T_E^0 a_E + T_P^0 (a_P^0 - a_N - a_S - a_E - a_W) + b \tag{4.6}$$

$$a_P = a_P^0 - G_P \Delta x \Delta y \tag{4.7a}$$

$$a_P^0 = \rho c_p \frac{\Delta x \Delta y}{\Delta t} \tag{4.7b}$$

$$b = G_C \Delta x \Delta y \tag{4.7c}$$

This both formulations are correct for all the interior nodes but now we have to take care of the boundary nodes. We need to make our case of study as generic as possible, thus we are going to study all the boundary conditions for conduction, convection at any position and conduction-convection at the corners.

- **Stability criterion**

We need to note that the coefficients of T_P^0 can become negative and the *stability criterion* tells us that all coefficients need to be positive. Taking Eq. 4.7b and Eq. 4.6 we see that Δt have to be small enough so that:

$$a_P^0 > a_N + a_S + a_E + a_W \tag{4.8}$$

For the convergence of the solutions, if this condition is violated we could get physical unrealistic results. From Eq.4.8 and the coefficient definition for explicit case (Eq.(4.3-4.7) we get:

$$\rho c_p \frac{\Delta x \Delta y}{\Delta t} > \frac{k_n \Delta x}{(\partial x)_n} + \frac{k_s \Delta x}{(\partial x)_s} + \frac{k_e \Delta y}{(\partial x)_e} + \frac{k_w \Delta y}{(\partial x)_w} \tag{4.9}$$

$$\frac{\rho c_p}{\Delta t} > \frac{k_n}{\Delta y (\partial x)_n} + \frac{k_s}{\Delta y (\partial x)_s} + \frac{k_e}{\Delta x (\partial x)_e} + \frac{k_w}{\Delta x (\partial x)_w} \tag{4.10}$$

$$\Delta t < \rho c_p \left[\frac{k_n}{\Delta y (\partial x)_n} + \frac{k_s}{\Delta y (\partial x)_s} + \frac{k_e}{\Delta x (\partial x)_e} + \frac{k_w}{\Delta x (\partial x)_w} \right]^{-1} \tag{4.11}$$

Equation 4.11 shows us that making the control-volumes smaller to improve the space accuracy implies reducing the time step Δt. In the following section we are going to see that the coefficients a_N, a_S, a_E, a_W change for the boundary nodes so the formulation of the *stability criterion* in coefficient form would be useful:

$$\Delta t < \frac{\rho c_p}{a_N + a_S + a_E + a_W} \tag{4.12}$$

Boundary Conditions for Explicit Formulation

First of all we are going to pose a case where the walls are surrounded by a solid material with conduction boundaries over all the faces of the two-dimensional plate and the same for convection boundaries but with a fluid surrounding the plate. Now we can start the study of the explicit formulation coefficients variation for every possible boundary node.

- **Conduction boundaries**

We know the coefficient form for the explicit formulation from Eq.4.6. For this study we are going to pose a *heat balance* to see what coefficients vary. From Appendix D Eq.D.5 and making reference to a South boundary node as Fig.D.1 shows, we can say

$$k_w \Delta y \frac{T_W^0 - T_P^0}{(\partial x)_w} + k_e \Delta y \frac{T_E^0 - T_P^0}{(\partial x)_E} + k_n \Delta x \frac{T_N^0 - T_P^0}{(\partial x)_n} + \left[k_P \Delta x \frac{T_{w1} - T_P^0}{(\partial x)_s / 2} \right] + G \Delta x \Delta y = \rho c_p \frac{\Delta x \Delta y}{\Delta t} (T_P^1 - T_P^0) \tag{4.13}$$

	Coefficient				**Temperature**			
Position	a_W	a_N	a_E	a_S	T_W^0	T_N^0	T_E^0	T_S^0
Wall 1	$\frac{k_w \Delta y}{(\partial x)_w}$	$\frac{k_n \Delta x}{(\partial x)_n}$	$\frac{k_e \Delta y}{(\partial x)_e}$	$\frac{k_P \Delta x}{(\partial x)_s/2}$	T_W	T_N	T_E	T_{w1}
Wall 2	$\frac{k_P \Delta y}{(\partial x)_w/2}$	$\frac{k_n \Delta x}{(\partial x)_n}$	$\frac{k_e \Delta y}{(\partial x)_e}$	$\frac{k_s \Delta x}{(\partial x)_s}$	T_{w2}	T_N	T_E	T_S
Wall 3	$\frac{k_w \Delta y}{(\partial x)_w}$	$\frac{k_P \Delta x}{(\partial x)_n/2}$	$\frac{k_e \Delta y}{(\partial x)_e}$	$\frac{k_s \Delta x}{(\partial x)_s}$	T_W	T_{w3}	T_E	T_S
Wall 4	$\frac{k_w \Delta y}{(\partial x)_w}$	$\frac{k_n \Delta x}{(\partial x)_n}$	$\frac{k_P \Delta y}{(\partial x)_e/2}$	$\frac{k_s \Delta x}{(\partial x)_s}$	T_W	T_N	T_{w4}	T_S
Corner 1	$\frac{k_P \Delta y}{(\partial x)_w/2}$	$\frac{k_n \Delta x}{(\partial x)_n}$	$\frac{k_e \Delta y}{(\partial x)_e}$	$\frac{k_P \Delta x}{(\partial x)_s/2}$	T_{w2}	T_N	T_E	T_{w1}
Corner 2	$\frac{k_P \Delta y}{(\partial x)_w/2}$	$\frac{k_P \Delta x}{(\partial x)_n/2}$	$\frac{k_e \Delta y}{(\partial x)_e}$	$\frac{k_s \Delta x}{(\partial x)_s}$	T_{w2}	T_{w3}	T_E	T_S
Corner 3	$\frac{k_w \Delta y}{(\partial x)_w}$	$\frac{k_P \Delta x}{(\partial x)_n/2}$	$\frac{k_P \Delta y}{(\partial x)_e/2}$	$\frac{k_s \Delta x}{(\partial x)_s}$	T_W	T_{w3}	T_{w4}	T_S
Corner 4	$\frac{k_w \Delta y}{(\partial x)_w}$	$\frac{k_n \Delta x}{(\partial x)_n}$	$\frac{k_P \Delta y}{(\partial x)_e/2}$	$\frac{k_P \Delta x}{(\partial x)_s/2}$	T_W	T_N	T_{w4}	T_{w1}

Table 4.4: Coefficients and temperatures for boundary two-dimensional conduction conditions

The only term that changes is the one that indicates heat flux over the South face of the control volume, extracting that conclusion we can intuit the coefficients and temperatures for any boundary position.

- **Convection boundaries**

For this case we need to imagine that the two-dimensional plate is surrounded by a fluid that may be at different temperatures for each wall. We are going to express the heat transfer by convection with the Newton's Law of Cooling[1]

$$q_{conv} = h(T_g w - T_P) \qquad h\left[\frac{W}{m^2 K}\right] \tag{4.14}$$

were T_{gw} is the fluid temperature at the wall, for simplification of the formulation we are going to call it T_{wi} as in the previous case. The term h is the heat transfer coefficient.

Once we know that we can propose the energy balance for the South boundary of our plate as Fig.**??** shows, getting

$$k_w \Delta y \frac{T_W^0 - T_P^0}{(\partial x)_w} + k_e \Delta y \frac{T_E^0 - T_P^0}{(\partial x)_E} + k_n \Delta x \frac{T_N^0 - T_P^0}{(\partial x)_n} + \left[h_1 \Delta x (T_{w1} - T_P\right] + G\Delta x \Delta y = \rho c_p \frac{\Delta x \Delta y}{\Delta t}(T_P^1 - T_P^0) \tag{4.15}$$

In Eq.4.15 we can see the same behavior as in Eq.4.13 since the only coefficient from the explicit formulation that change is the one related to the boundary study, in this first case a_S. Using this creterion we can write a new coefficient table which looks like to Table.4.4 but with some changes related to the conduction boundaries.

[1] When the temperature difference between a body and its environment is not too great, the heat transferred in the unit of time to the body or from the body by conduction, convection and radiation is approximately proportional to the temperature difference between the body and the external environment [15]

	Coefficient				Temperature			
Position	a_W	a_N	a_E	a_S	T_W^0	T_N^0	T_E^0	T_S^0
Wall 1	$\frac{k_w \Delta y}{(\partial x)_w}$	$\frac{k_n \Delta x}{(\partial x)_n}$	$\frac{k_e \Delta y}{(\partial x)_e}$	$h_1 \Delta x$	T_W	T_N	T_E	T_{w1}
Wall 2	$h_2 \Delta y$	$\frac{k_n \Delta x}{(\partial x)_n}$	$\frac{k_e \Delta y}{(\partial x)_e}$	$\frac{k_s \Delta x}{(\partial x)_s}$	T_{w2}	T_N	T_E	T_S
Wall 3	$\frac{k_w \Delta y}{(\partial x)_w}$	$h_3 \Delta x$	$\frac{k_e \Delta y}{(\partial x)_e}$	$\frac{k_s \Delta x}{(\partial x)_s}$	T_W	T_{w3}	T_E	T_S
Wall 4	$\frac{k_w \Delta y}{(\partial x)_w}$	$\frac{k_n \Delta x}{(\partial x)_n}$	$h_4 \Delta y$	$\frac{k_s \Delta x}{(\partial x)_s}$	T_W	T_N	T_{w4}	T_S
Corner 1	$h_2 \Delta y$	$\frac{k_n \Delta x}{(\partial x)_n}$	$\frac{k_e \Delta y}{(\partial x)_e}$	$h_1 \Delta x$	T_{w2}	T_N	T_E	T_{w1}
Corner 2	$h_2 \Delta y$	$h_3 \Delta x$	$\frac{k_e \Delta y}{(\partial x)_e}$	$\frac{k_s \Delta x}{(\partial x)_s}$	T_{w2}	T_{w3}	T_E	T_S
Corner 3	$\frac{k_w \Delta y}{(\partial x)_w}$	$h_3 \Delta x$	$h_4 \Delta y$	$\frac{k_s \Delta x}{(\partial x)_s}$	T_W	T_{w3}	T_{w4}	T_S
Corner 4	$\frac{k_w \Delta y}{(\partial x)_w}$	$\frac{k_n \Delta x}{(\partial x)_n}$	$h_4 \Delta y$	$h_4 \Delta x$	T_W	T_N	T_{w4}	T_{w1}

Table 4.5: Coefficients and temperatures for boundary two-dimensional convection conditions

As we have seen in the study of convection and conduction boundary conditions, this phenomenons only affects to a_N, a_W, a_S, a_E coefficients and their corresponding temperatures.

4.1.4 Explicit Algorithm

The explicit formulation (Eq.4.6) is now ready to be applied in our solver for a two-dimensional heat conduction problem because all the coefficients are known for interior and boundary nodes. We have already noticed that there is only one unknown term (T_P) which can be easily separated to the left side of the equation

$$T_P = \frac{1}{a_P}\Big[T_N^0 a_N + T_S^0 a_S + T_W^0 a_W + T_E^0 a_E + T_P^0(a_P^0 - a_N - a_S - a_E - a_W) + b\Big] \tag{4.16}$$

which leads to a good implementation of the *Point-By-Point* method explained in Section 3.7.1. Besides, we need to follow the *stability criterion* from Eq.4.12 for each node of the discretization. Explicit method gives us the value of T_P directly without the implementation of a iterative solver since there are no guessed values of temperature.

4.2 Convection-Diffusion Solenoidal Flow Problem

In this section it is going to be developed the concepts explained in Section 2.3 applied to a practical case where the convection-diffusion phenomena is involved proposed by the CTTC. This problem can also be called Smith-Hutton Problem.

This is a recirculating flow problem which involves streamline curvature studied by Smith and Hutton. In their study they concluded that in a high-convection regime modelling "remains the art of compromise between diffusive and oscillatory errors".[16]

4.2.1 Problem Definition

This problem is based in a two-dimensional test problem devised by Smith and Hutton which concerns steady-state convection and diffusion of a scalar field ϕ in a prescribed velocity field $\overrightarrow{v}$ with a known constant diffusivity D. The objective is to find the field ϕ for a given value of the relation ϕ/Γ. Figure 4.2 shows a visual scheme of the problem.

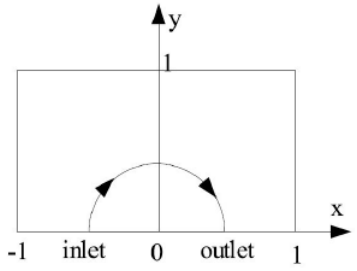


Figure 4.2: Smith-Hutton problem

As we can see in Fig.4.2 the flow domain considered is a rectangle: $-1 \leq x \leq 1$, $0 \leq y \leq 1$. And the velocity field $\overrightarrow{v}$ is given by

$$u(x,y) = 2y(1-x^2) \tag{4.17a}$$

$$v(x,y) = -2x(1-y^2) \tag{4.17b}$$

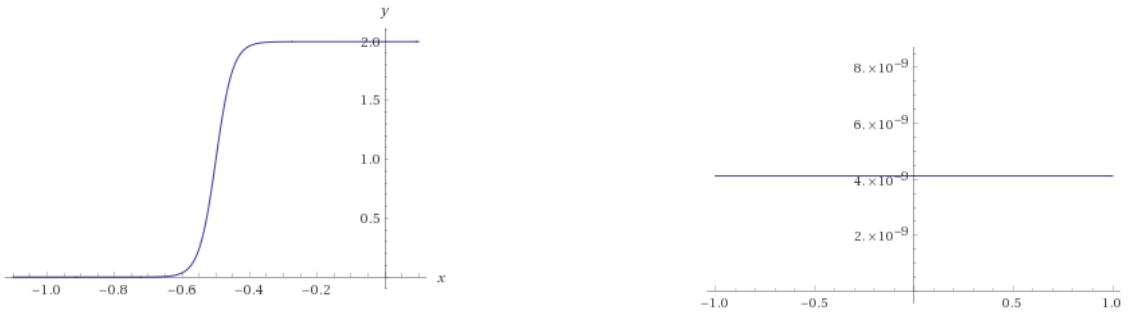
4.2.2 Boundary Conditions

Table 4.6 gives us the boundary conditions for the parameter ϕ in our case of study.

Field ϕ	$x[m]$	$y[m]$
$\phi = 1 + tanh[(2x+1)\alpha]$	$-1 < x < 0$	$y = 0$
	$x = -1$	$0 < y < 1$
$\phi = 1 + tanh(\alpha)$	$-1 < x < 1$	$y = 1$
	$x = 1$	$0 < y < 1$
$d\phi/dy = 0$	$0 < x < 1$	$y = 0$

Table 4.6: Boundary conditions for Smith-Hutton Problem ($\alpha = 10$)

As we can see in the plots shown in Fig. 4.3 the hyperbolic tangent function[2] in the inlet boundary condition ($-1 < x < 0$, $y = 0$) gives a values of ϕ almost 0 for $-1 < x < -0.5$ and rapidly grows to a 2 value for $-0.5 < x < 0$. For all the other boundaries the value of $\phi$is approximated to 0.



(a) (b)

Figure 4.3: Plots for the hyperbolic tangent functions (a)Inlet (b)No flow boundaries

4.2.3 Discretization

From Section 2.3.2 we can rewrite the convection-diffusion equation (Eq.2.28) considering the source term value as zero:

$$\rho\frac{d\phi}{dt} + \rho u\frac{d\phi}{dx} + \rho v\frac{d\phi}{dy} = \frac{k}{c_p}\Big(\frac{d^2\phi}{dx^2} + \frac{d^2\phi}{dy^2}\Big) \tag{4.18}$$

The implicit coefficient form of the previous equation is known from Section 2.3.5

$$a_P\phi_P = a_E\phi_E + a_S\phi_S + a_W\phi_W + a_N\phi_N + b \tag{4.19}$$

Even though our problem asks for the steady state condition, the terms that contain time dependency were taken into account because we can obtain more conclusions about time evolution of the phenomena and his computational cost. The scheme chosen to solve the convection-diffusion equation is a implicit scheme because it gives physically satisfactory results . The spatial discretization and control volume geometry chosen for our case of study are shown in Table 4.7.

Spatial Property	L	H	N_x	N_y	Δx	Δy
Value	$[-1, 1]$	1	ins	ins	ins	ins

Table 4.7: Domain spatial discretization for Smith-Huton Problem

[2]Smith and Hutton proposed $\alpha = 10$ as representative of a relatively sharp transmission[16]

In Section 3.4 it is said that the mesh used for this thesis is the Grid B shown in Fig.3.3. In this disposal of the grid points its important to note that there are grid nodes around all the boundary conveniently located at wall faces of the control volume. Using this Grid allows the direct determination of boundary conditions and an easier analysis of the coefficients at this points. It is important to take into account that the distance between between this nodes and its neighbors is half of the central grid nodes.

4.2.4 Algorithm

With the purpose of understanding the algorithm developed for solving this problem Fig.4.4 gives us an idea about the main processes inside the code. For this problem a Matlab Object Oriented code has been developed as a first contact with this programming method. The core of the code is the Main function, which contains all the needed methods and input data. Inside the main we can find four principal functions:

- Uniform Mesh: in charge of the domain discretization and compute of the velocity field needed for each problem.
- Coefficient Compute: in charge of computing the needed coefficients for each case, the ones that are dependent on the field ϕ and the non-dependent.
- Solver: it is in charge of finding the value of ϕ for the coefficients previously calculated.
- Solver Shell: that function returns us the final results for the ϕ field. It contains the Solver.

The needed data for starting the computations is modified from the "inputData" file. It contains the points that define our domain (P_1, P_2), the requested solutions points, the sizes of our mesh (N_x, N_y), the initial field ϕ value and his boundary conditions, the physical properties needed for solving the coefficients and finally the solver parameters.

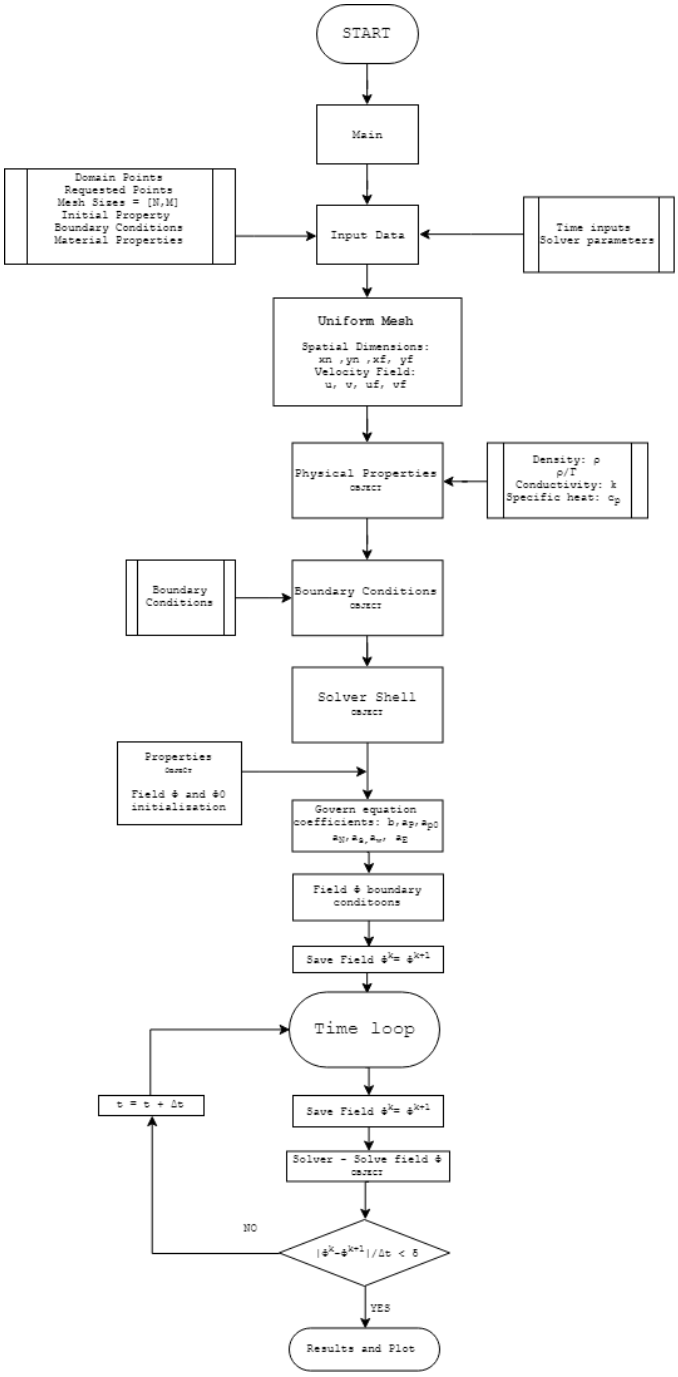


Figure 4.4: Smith-Hutton Problem algorithm flowchart

4.2.5 Results

Once our code is running and working correctly it is needed to compare the results obtained at the outlet of our contour with numerical results provided by [4] in order to check their validity. The results are displayed in Table 4.8 and we can see the complete field for each situation in the plots shown below (Fig. **??-??**).
Explicar esquema implementat i solver utilitzat per als resultats

	$\rho/\Gamma = 10$		$\rho/\Gamma = 1000$		$\rho/\Gamma = 10$	
Position x	Expected	Calculated	Expected	Calculated	Expected	Calculated
0.0	1.989		2.0000		2.000	
0.1	1.402		1.9990		2.000	
0.2	1.146		1.9997		2.000	
0.3	0.946		1.9850		1.999	
0.3	0.775		1.8410		1.000	
0.5	0.621		0.9510		0.036	
0.6	0.480		0.1546		0.001	
0.7	0.349		0.0010		0.000	
0.8	0.227		0.0000		0.000	
0.9	0.111		0.0000		0.000	
1.0	0.000		0.0000		0.000	

Table 4.8: Numerical results at the outlet for different ρ/Γ [4]

4.2.6 Conclusions

As the ρ/Γ ratio increases, the convective term grows taking a predominant role against the diffusive term, wich decreases. This behaviour can be observed in the field ϕ plots shown in the figures below. In the first case where $\rho/\Gamma = 10$ we know from the Peclet Eq. 2.8 that for this values the Peclet number is low. Which means that for low Peclet numbers the problem tens to have greater diffusive effects. But as Peclet number increases the convective term gains influence, for that reason a solenoidal field is observed for the cases of greater ϕ/Γ.

Chapter 5

Economic and Enviromental Study

Contents

5.1 Economics and Budget

5.2 Enviromental impact

Chapter 6

Conclusions and Future Actions

Contents

6.1 Conclusions

6.2 Future Actions

Chapter 7

Task Planning

Contents

7.1 Task Description

7.2 Future Tasks Estimation

References

[1] S. V.Patankar, *Numerical Heat Transfer and Fluid Flow*, 1st ed., 1980.

[2] D. M. Strong, "Iterative methods for solving ax = b - jacobi's method." [Online]. Available: https://www.maa.org/press/periodicals/loci/joma/iterative-methods-for-solving-iaxi-ibi-jacobis-method

[3] Y. A. Cengel, *Heat transfer: a practical approach*, 2nd ed., 2004.

[4] D. CTTC, *Validation of the Convection-Diffusion Equation, Tech. rep. , ESEIAAT.*

[5] K. W. Morton, *Numerical solution of convection-diffusion problems*, 1st ed., 1996.

[6] F. M. White, *Fluid Mechanics*, 5th ed.

[7] D. CTTC, *Numerical solution of convection. Tech. rep. , ESEIAAT*, 2010.

[8] J. H. Ferziger and M. Peric, *Computational Methods for Fluid Dynamics*, 3rd ed. Springer, 2002.

[9] G. Dahlquist and A. Bjorck, *Numerical Methods*, 1st ed. Penitence Hall, 1969.

[10] K. A. Atkinson, *An introduction to Numerical Analysis*, 2nd ed. John Wiley Sons, 1989.

[11] D. Bindel and J. Goodman, *Principles of Scientific Computing Linear Algebra II, Algorithms, NYU*, 2009.

[12] Vaibhavrai3, "Lu decomposition of a system of linear equations." [Online]. Available: https://www.geeksforgeeks.org/l-u-decomposition-system-linear-equations/

[13] S. Mazumder, *Numerical Methods for Partial Differential Equations: Finite Difference and Finite Volume Methods. Academic Press*, 2015.

[14] D. M. Strong, "Iterative methods for solving ax = b - gauss-seidel method." [Online]. Available: https://www.maa.org/press/periodicals/loci/joma/iterative-methods-for-solving-iaxi-ibi-gauss-seidel-method

[15] Sc.ehu.es, "Ley del enfriamiento de newton." [Online]. Available: http://www.sc.ehu.es/sbweb/fisica/estadistica/otros/enfriamiento/enfriamiento.html

[16] B. Leonard and S. Mokhtari, *ULTRA-SHARP Solution of the Smith-Hutton Problem.* Department of Mechanical Engineering, The University of Akron, 1992.

[17] H. K. Varsteeg and W. Malalasekera, *An Introduction to Computational Fluid Dynamics: The Finite Volume Method*, 2007.

[18] J. B. Scarborough, *Numerical Mathematical Analysis*, 1st ed., 1955.

Appendix A

One-Dimensional Conduction Govern Equation

[3]Considering a onde-dimensional control-volume of thickness Δx as shown in Fig.A.1 an *energy balance* on this thin element during a small interval of time Δt can be expressed as

$$\left(\begin{array}{c}\text{Rate of heat}\\ \text{conduction}\\ \text{at } x\end{array}\right) - \left(\begin{array}{c}\text{Rate of heat}\\ \text{conduction}\\ \text{at } x+\Delta x\end{array}\right) + \left(\begin{array}{c}\text{Rate of heat}\\ \text{generation}\\ \text{inside the}\\ \text{element}\end{array}\right) = \left(\begin{array}{c}\text{Rate of change}\\ \text{of the energy}\\ \text{content of the}\\ \text{element}\end{array}\right)$$

or

$$\dot{Q}_x - \dot{Q}_{x+\Delta x} + \dot{G}_{CV} = \frac{\Delta E_{CV}}{\Delta t} \tag{A.1}$$

Subindex CV means control-volume. Change in energy content and heat generation can be expressed as

$$\Delta E_{CV} = E_{t+\Delta t} - E_t = mc_p(T_{t+\Delta t} - T_t) = \rho c_p A \Delta (T_{t+\Delta t} - T_t) \tag{A.2}$$

$$\dot{G}_{CV} = GV_{CV} = GA\Delta x \tag{A.3}$$

Subtituting into Eq.A.1, we get

$$\dot{Q}_x - \dot{Q}_{x+\Delta x} + GA\Delta x = \rho c_p A \Delta x \frac{(T_{t+\Delta t} - T_t)}{\Delta t} \tag{A.4}$$

Dividing by $A\Delta x$ gives

$$-\frac{1}{A}\frac{\dot{Q}_x - \dot{Q}_{x+\Delta x}}{\Delta x} + G = \rho c_p \frac{(T_{t+\Delta t} - T_t)}{\Delta t} \tag{A.5}$$

Taking the limit as $\Delta x \to 0$ and $\Delta t \to 0$ yields[1]

$$\frac{1}{A}\frac{d}{dx}\left(kA\frac{dT}{dx}\right) + G = \rho c_p \frac{dT}{dt} \tag{A.7}$$

Noting that the area A is constant for our case of study, as Fig.A.1 shows, the one-dimensional transient heat conduction equation becomes

$$\frac{d}{dx}\left(k\frac{dT}{dx}\right) + G = \rho c_p \frac{dT}{dt} \tag{A.8}$$

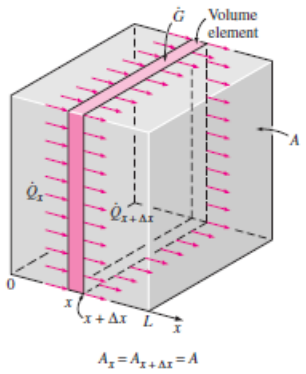


Figure A.1: One-dimensional heat conduction through a volume element[3]

[1] From the definition of derivative and Fourier's law of heat conduction

$$\lim_{x\to 0}\frac{\dot{Q}_{x+\Delta x} - \dot{Q}_x}{\Delta x} = \frac{d\dot{Q}}{dx} = \frac{d}{dx}\left(-kA\frac{dT}{dx}\right) \tag{A.6}$$

Appendix B

General One-Dimensional Discretization Equation Solving

$$\frac{d}{dx}\Big(k\frac{dT}{dx}\Big) = \rho c_p \frac{dT}{dt} \tag{B.1}$$

Integrating Eq.B.1:

$$\rho c_p \int_w^e \int_t^{t+\Delta t} \frac{dT}{dt} dt dx = \int_t^{t+\Delta t} \int_w^e \frac{d}{dx}\Big(k\frac{dT}{dx}\Big) dx dt \tag{B.2}$$

Where:

$$\rho c_p \int_w^e \int_t^{t+\Delta t} \frac{dT}{dt} dt dx = \rho c_p \Delta x (T_P^1 - T_P^0) \tag{B.3}$$

$$\int_w^e \frac{d}{dx}\Big(k\frac{dT}{dx}\Big) dx = \frac{k_e(T_E - T_P)}{(\partial x)_e} + \frac{k_w(T_W - T_P)}{(\partial x)_w} \tag{B.4}$$

Introducing equations B.3 and B.4 into Eq. B.2 we get

$$\rho c_p \Delta x (T_P^1 - T_P^0) = \int_t^{t+\Delta t} \left[\frac{k_e(T_E - T_P)}{(\partial x)_e} + \frac{k_w(T_W - T_P)}{(\partial x)_w}\right] dt \tag{B.5}$$

It is at this point when we need an assumption about how T_P T_E and T_W varies with time from t to $t + \Delta t$. Many are possible and in a generalized form we can state:

$$\int_t^{t+\Delta t} T dt = [f T_P^1 + (1-f) T_P^0] \tag{B.6}$$

where f is a weighting factor between 0 and 1 that defines the assumption done in the previous step. From Eq. B.5 using Eq.B.6 assumption we obtain:

$$\rho c_p \frac{\Delta x}{\Delta t}(T_P^1 - T_P^0) = \\ = f\left[\frac{k_e(T_E^1 - T_P^1)}{(\partial x)_e} + \frac{k_w(T_W^1 - T_P^1)}{(\partial x)_w}\right] + (1-f)\left[\frac{k_e(T_E^0 - T_P^0)}{(\partial x)_e} + \frac{k_w(T_W^0 - T_P^0)}{(\partial x)_w}\right] \tag{B.7}$$

Appendix C

Interface Conductivity Energy Balance

Figure C.1 refers to the composite slab steady one dimensional analysis without heat sources. For finding a generic solution to this problem it is needed to propose an energy balance over the interface. As the temperature over the interface is constant we can state that the heat flux over the interface is constant:

$$q_e^P = q_e^E \tag{C.1}$$

From Eq.2.12:

$$q_e = \frac{k_P(T_P - T_e)}{(\partial x)_{e-}} = \frac{k_E(T_e - T_E)}{(\partial x)_{e+}} \tag{C.2}$$

Taking only T_e to the left side of the equation

$$T_e = \frac{T_P \dfrac{k_P}{(\partial x)_{e-}} + T_E \dfrac{k_E}{(\partial x)_{e+}}}{\dfrac{k_P}{(\partial x)_{e-}} + \dfrac{k_E}{(\partial x)_{e+}}} \tag{C.3}$$

Introducing Eq.C.3 into second term of Eq.C.2:

$$q_e = \frac{k_P}{(\partial x)_{e-}} \left(T_P - \frac{T_P \dfrac{k_P}{(\partial x)_{e-}} + T_E \dfrac{k_E}{(\partial x)_{e+}}}{\dfrac{k_P}{(\partial x)_{e-}} + \dfrac{k_E}{(\partial x)_{e+}}} \right) \tag{C.4}$$

After reordering terms and simplifying we obtain the desired form of the equation:

$$q_e = \frac{k_P k_E}{k_P(\partial x)_{e+} + k_E(\partial x)_{e-}}(T_P - T_E) = \frac{T_P - T_E}{\frac{(\partial x)_{e+}}{k_E} + \frac{(\partial x)_{e-}}{k_P}} \tag{C.5}$$

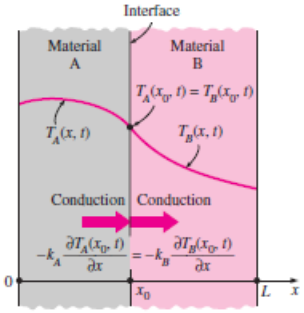


Figure C.1: Boundary conditions at the interface [3]

Appendix D

Two-Dimensional Conduction Heat Balance

We are going to consider the heat transfer to be into the volume element at all surfaces of the control volume as the Fig.**??** shows. The assumed direction of heat transfer at surfaces of the control volume has no effect on the *finite difference* formulation but it has at the time of organizing the coefficients of the final form of the equation.

$$Q_E + Q_W + Q_N + Q_S + \dot{G} = \rho c_p \frac{\Delta x \Delta y}{\Delta t}(T_P^1 - T_P^0) \tag{D.1}$$

$$k_w A_w \frac{T_W - T_P}{(\partial x)_w} + k_e A_e \frac{T_E - T_P}{(\partial x)_E} + k_n A_n \frac{T_N - T_P}{(\partial x)_n} + k_s A_s \frac{T_S - T_P}{(\partial x)_s} + \dot{G} = \rho c_p \frac{\Delta x \Delta y}{\Delta t}(T_P^1 - T_P^0) \tag{D.2}$$

For our problem conditions:

$$A_w = A_e = \Delta y \tag{D.3}$$

$$A_n = A_s = \Delta x \tag{D.4}$$

Introducing equations (D.3-D.4) into Eq.D.2

$$k_w \Delta y \frac{T_W - T_P}{(\partial x)_w} + k_e \Delta y \frac{T_E - T_P}{(\partial x)_E} + k_n \Delta x \frac{T_N - T_P}{(\partial x)_n} + k_s \Delta x \frac{T_S - T_P}{(\partial x)_s} + G \Delta x \Delta y = \rho c_p \frac{\Delta x \Delta y}{\Delta t}(T_P^1 - T_P^0) \tag{D.5}$$

Rearranging the terms we obtain the coefficient equation

$$a_W T_W + a_E T_E + a_N T_N + a_S T_S - T_P(a_W + a_E + a_N + a_S) + G\Delta x\Delta y = \rho c_p \frac{\Delta x \Delta y}{\Delta t}(T_P^1 - T_P^0) \quad \text{(D.6)}$$

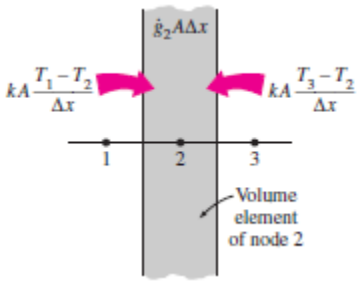


Figure D.1: Heat transfer to be into a volume element[3]

Appendix E

Scarborough Criterion

In this Appendix we can find the definition of the Scarborough criterion and it's mathematical formulation. This criterion is used for satisfying the convergence of the solution of linear equation systems in iterative problems. It can be expressed in terms of the values of the coefficients of the discretised equations:

$$\frac{\Sigma|a_{ib}|}{|a_P|} \left\{ \begin{array}{ll} \leq 1, & \text{at all nodes} \\ < 1, & \text{at one node at least} \end{array} \right\} \tag{E.1}$$

Where a_p corresponds to a random grid point P and the numerator sum states for the sum of all neighbour coefficients to node P. This is a sufficient condition but not a necessary one, what means that we can get convergence even if we violate the criterion at some time and the satisfaction of this criterion ensures that the equations will be converged by at least one iterative method. For more information see [17, 18, 1].