**POLITECHNIKA WARSZAWSKA**

**Faculty of Power and Aeronautical Engineering**

**Master's Thesis**

# RESOLUTION OF THE NAVIER STOKES EQUATION IN LAMINAR AND TURBULENT REGIMES

**Ezekiel Atiba (302557)**

**Exchange Student at Heat and Mass Transfer Technological Center (CTTC), Universitat Politècnica de Catalunya, Terrassa, Spain.**

Centre Tecnològic de Transferència de Calor
UNIVERSITAT POLITÈCNICA DE CATALUNYA

**Supervisor: Prof David Perez Segarra**

**Terrassa 2020**

**ABSTRACT**

The main aim of this project is to develop numerical codes that solve basic fluid dynamics and heat transfer problems. Following the finite volume method, the respective partial differential equations for the five problems considered are solved. Requisite comparisons with literature and benchmark results with own simulations of the problems are presented. The five cases considered are pure conduction, convection-diffusion, driven cavity, differentially heated cavity, and burgers equation.

**ACKNOWLEDGEMENT**

Special thanks to Prof. Carles-David Perez Segarra and Prof. Asensi Oliva Llena, the supervisor and co-supervisor of this thesis, for their immense support and leadership.

I also appreciate dr hab. inż. Piotr Łapka, the supervisor at my home University.

I register my utmost appreciation to Jordi Chiva, Dr. Deniz Kizildag, Dr. Francesc Xavier Trias and Adel Alsalti for the tutorials explanation and guidance throughout the course of this work. Amazing!

**TABLE OF CONTENT**

## LIST OF TABLES

**LIST OF FIGURES**

# CHAPTER ONE

## 1.0 INTRODUCTION

### The Navier-Stokes Equation

Most engineering problems are defined by heat transfer and/or fluid flow. In power generation, heat transfer and fluid flow play essential roles in the design and operation of devices, processes and system. In aerospace, heat transfer is responsible for the expansion of air-fuel mixture in the turbojet while fluid flow describes the effect of flow around the body of the aircraft as it moves in the air. Heat transfer and fluid flow are important in other areas such as environment and weather conditions, metallurgy, heating and air conditioning, electronics and electric machinery and a host of other applications.

The motion of fluid is governed by a system of Partial Differential Equations known as the Navier-Stokes equation. This system of equation is deduced from the principles of mass conservation, momentum conservation, energy conservation, second law of thermodynamics and the definition of a fluid. Given some initial conditions, the Navier-stokes equation can be used to obtain the velocity field of a fluid. It should be stated that the Navier-Stokes Equation arise from the application of Newton's second law in combination with a fluid stress (due to viscosity) and a pressure term.

For an incompressible flow, which shall be the focus of this project, the Navier-stokes equation is given by

$$\frac{\partial \mathbf{v}}{\partial t} + (\mathbf{v}\cdot\nabla)\mathbf{v} = \frac{1}{Re}\Delta\mathbf{v} - \nabla P$$

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

From the equation above, the first, second, third and fourth terms describe the unsteady, convective, diffusive and the pressure terms respectively. The change in the diffusive term, by the instrument of the Reynolds number determines whether the flow is turbulent of Laminar.

The Navier-Stokes equation possess a rather complex nature; solving analytically could be difficult or nearly impossible. Therefore, simulation and approximations are employed

to utilize the Navier-stokes equation. This form the basis for Computational fluid dynamics and Numerical methods.

### Aim of the Thesis

The aim of this study is to gain understand into the physics behind the fluid dynamics and heat transfer as well as the development, validaton and verification own codes for solving computational fluid dynamics (CFD) and heat transfer (HT) problems

### Scope

The scope of this project is a stepwise and systematic one. Four cases, one for each phenomenon, involving fluid flow and heat transfer are simulated. The governing Partial differential equations (PDEs) for each case is discretized using the Finite Volume Method, after which it is implemented into a C++ program. The results obtained are used for some number of comparative analysis.

The following phenomena are considered;

- **Pure Conduction:** Transient heat conduction in composite wall
- **Convection-Diffusion:** Smith Hutton
- **Driven Cavity:** Forced convection
- **Differentially Heated Cavity:** Natural Convection
- **Burgers Equation:** Introduction to Turbulence

# CHAPTER TWO

## 2.0 BACKGROUND STUDIES

### 2.1 NUMERICAL METHODS

Numerical methods are mathematical methods that are used to approximate the solution of complicated problems so that the solution consists of only addition, subtraction and multiplication operations. Computational fluid dynamics (CFD), which is a special kind of numerical analysis is done to understand fluid flow patterns in fluid machinery system. Numerical methods and CFD involves the following aspects;

- Mathematical model
- Discretization process
- Solution Method
- Convergence and Stability
- Post-processing and interpretation

### 2.2 MATHEMATICAL MODEL

According to [1], A mathematical model can be broadly defined as a formulation or equation that expresses the essential features of a physical system or process in mathematical terms. Through the mathematical model, we can establish the governing equations that describes the process of interest. As Patankar [2] puts it, the numerical solution of heat transfer, fluid flow, and other related processes can begin when laws governing these processes have been expressed in mathematical form, generally differential equations. In the case of fluid flow problem, the Navier-Stokes equation, which has been described in the introductory part of the thesis, is the mathematical model. It should be noted that the differential equations express conservation principles where there is a balance in the various factors that influence the dependent variable of the physical quantity being solved.

## 2.3 DISCRETIZATION PROCESS

Since it has been established that it is rather impossible to employ analytical method to solve the Partial differential equations (PDEs) that describe fluid flow and related phenomena, the discretization process offers a way to obtain an approximate solution. The governing PDEs are reduced to a set of algebraic equations which can then be solved on a computer. As it is stated in [3], the numerical solution of a partial differential equation consists of finding the values of the dependent variable $\phi$ at specified points from which its distribution over the domain of interest can be constructed. Generally, the discretization process involves two steps; domain discretization and equation discretization.

DOMAIN DISCRETIZATION

Arriving at an approximate solution for the value of any dependent variable of interests starts with discretizing the physical domain into discrete points otherwise known as nodes. This allows algebraic equation to be formulated over the domain. The result of the geometric discretization of the physical is known as mesh. The mesh is composed of discrete elements defined by sets of vertices and bounded by faces.

To achieve domain discretization, various approaches have been utilized by scientists and researchers. The three most popular methods are Finite Difference Method, Finite Volume Method, Finite Element Method.

### Finite Difference Method

The Finite difference method (FDM) works by replacing the derivatives of governing PDE with finite (grid), algebraic differences quotients. The partial derivatives in the PDE at each grip point (fig. 1) are approximated from neighbouring values using the Taylor's series[4]. This results in one algebraic equation per grid node, in which the variable value at that and a certain number of neighbour nodes appear as unknowns.

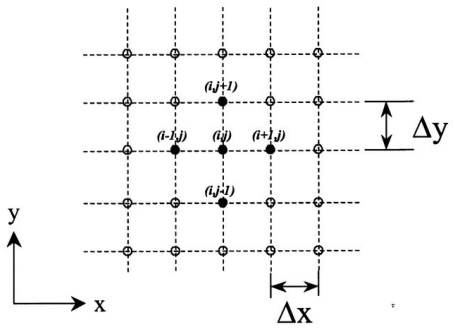



*Figure 1: FDM grid arrangement*

Usually, the Taylor's series is truncated after 1 or 2 terms, however, if more accuracy is desired, the number of terms may be increased.

Both structured and unstructured types mesh are applicable in the FDM. However, the structured mesh, offers more straightforwardness whereas, the unstructured mesh requires some transformations in the equation.

## Finite Volume Method

It is considered the generic conservation equation for a quantity ø and assume that the velocity field and all fluid properties are known[1]. The finite volume method (FVM) uses the integral form of the conservation equation as the starting point

$$\int_s \rho\phi \cdot nds = \int_s \Gamma\nabla\phi\cdot\mathrm{nds} + \int_\Omega q_\phi d\Omega$$

Finite volume method is a method for representing and evaluating partial differential equations in the form of algebraic equations. In the finite volume method, volume integrals in a partial differential equation that contain a divergence term are converted to surface integrals using the divergence theorem. It is useful for problems with body-fitted coordinate systems.

The control volume-based technique consists of the following;

- Division of the domain into discrete control volumes (CV) using a computational grid. The grid defines the boundaries of a control volume, while the computational node lies at the center of each control volume as shown in figure 2.
- Integration of the governing equations on the individual control volumes to construct algebraic equations for the discrete dependent variables ("unknowns'') such as velocities, pressure, temperature, and conserved scalars.
- Linearization of the discretized equations and solution of the resultant linear equation system to yield updated values of the dependent variables.

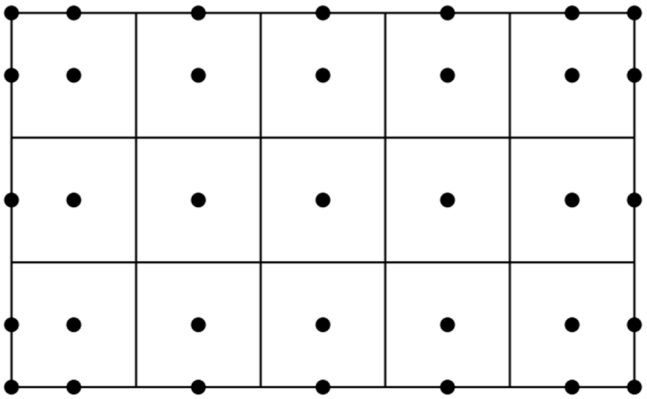


*Figure 2: FVM Grid Arrangement*

The FVM is the discretization approach that is being utilized in the course of this project, and more reference shall be made subsequently.

**Finite Element Method**

The FEM works on the choice of weight and shape function. The domain is subdivided into a finite number of elements, the governing equation is solved for each element and then overall solution is obtained by assembly. Galerkin's method of weighted residuals is generally used. The governing partial differential equations are integrated over an element or volume after having been multiplied by a weight function. The dependent variables are represented on the element by a shape function, which is the same form as the weight function[5]. The shape function may take any of several forms.

EQUATION DISCRETIZATION

Spatial Discretization

For the FVM and indeed other discretization method, the equation discretization step is performed over each Control volume (CV) to produce an algebraic relation that connect the value of a variable in a CV to the values of the variable in the neighbouring CV.

Given a control volume P in fig. 3, the discretized equation of the gradient of ø at face e is given by the following;

$$\left(\frac{\partial \emptyset}{\partial x}\right)_e = \frac{\emptyset_E - \emptyset_P}{\delta x_e}$$

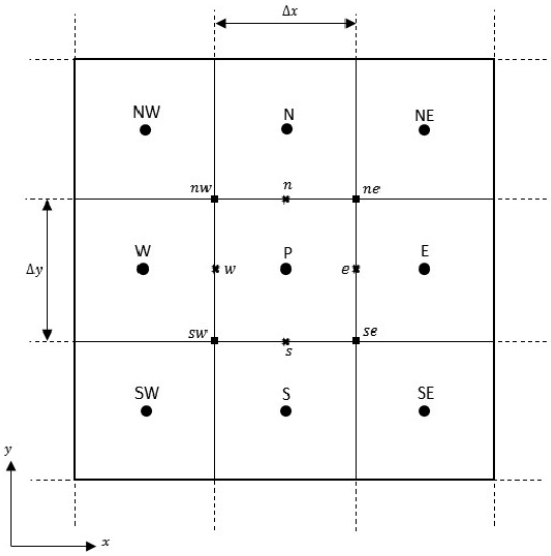



*Figure 3: Control Volume*

Analogously, this can be obtained for other faces of the CV and the corresponding coordinate.

Also, other differential operators such as divergence, and Laplacian can also be obtained from the spatial partial derivates of the field.

## Temporal Discretization

For unsteady cases, it is necessary to divide the temporal interval into discrete pieces. These pieces are known as time-steps. Temporal discretization involves the integration of every term in different equations over a time step. Starting with an initial condition at time $t = t_0$, the solution algorithm marches forward and finds a solution at time $t_1 = t_0 + \Delta t$. The solution found is the initial condition for the next time step and is used to obtain the solution at time $t_2 = t_1 + \Delta t$[3].

Therefore, just like the equation for spatial discretized equation, the temporal discretized equation for a control volume is obtained as follows;

$$\left(\frac{\partial \emptyset}{\partial t}\right)_{t_n} = g(\emptyset)$$

$$g(\emptyset) = \frac{{\emptyset_P}^{n+1} - {\emptyset_P}^{n}}{\Delta t}$$

A further step is taken in the evaluation of g(ø); the time level of ø is required. The three major schemes of choice are available.

**Implicit Time Integration**

This method evaluates g(ø) at the future level. The main principle of the implicit scheme is that the new value of $\emptyset_P$ prevails over the entire time step[2]. The function becomes

$$\frac{{\emptyset_P}^{n+1} - {\emptyset_P}^{n}}{\Delta t} = g(\emptyset^{n+1})$$

Though its computational cost is high, the fully implicit scheme advantageous in that it is unconditionally stable with respect to time step size.

**Explicit Time Integration**

The explicit scheme evaluated g(ø) at the current time. It is referred to as explicit because $\emptyset^{n+1}$ can be expressed explicitly in terms of the current value $\emptyset^{n}$.

The function becomes

$$\frac{{\emptyset_P}^{n+1} - {\emptyset_P}^{n}}{\Delta t} = g(\emptyset^{n})$$

It should be noted that there is a restriction to how large Δt can be while utilizing the explicit time integration. An inappropriate value could cause instability. To mitigate this, the Courant-Friedrich-Lewy condition is imposed.

The other scheme is the **Crank-Nicolson Scheme** which usually is unconditionally stable. For small time steps, it is more accurate than the implicit time integration.

$$g(\emptyset^{n+1}) = \frac{{\emptyset_P}^{n+1} - {\emptyset_P}^{n}}{2\Delta t}$$

## 2.4 SOLUTION METHOD

The discretization equations for each CV in the domain produce a set of algebraic equations. These algebraic equations relate the value of the dependent variable at the center of the CV to the neighbouring CVs. The discretization equations are in the form of a linear equation system, which are generally represented by

$$A[x]=b$$

Where the unknown variable x at the center of the CV and at the boundary of the domain. The number of CV determines the number of equations and unknowns. There are various

methods to arrive at the solution of this linear system of equation which are discussed next.

**Direct Methods**

The direct methods for approximating the solution of a system of 'n' linear equations in 'n' unknowns is one that gives the exact solution to the system, if it is assumed that all calculations can be performed without round-off error effects. This assumption however is idealized. Examples of direct methods include;

- Crammer's rule
- Gauss Elimination
- Pivoting
- LU Factorization

Despite the exactness and accuracy of direct methods, they are expensive in space and time.

**Iterative Methods**

The iterative techniques which require an initial approximation to the solution, will not be expected to return the exact solution even if all the calculations could be performed using exact arithmetic. In many instances, however, they are more effective than the direct methods, since they can require far less computational effort and round-off error is reduced. The types of iterative methods used in numerical methods are as follows

- Jacobi method
- Tri-diagonal matrix algorithm
- Line by line method
- Gauss-Seidel method

The gauss seidel method is utilized in this project. According to [2], The simplest of all iterative methods is the Gauss-Seidel method (GS) in which the values of the variable are calculated by visiting each grid point in a certain order. The property used next to explain this method is the temperature; nevertheless, it can be replaced to its general form with the property ø.

If the discretization equation is written as

$$a_P T_{P=} \sum a_{nb} T_{nb} + b$$

where the subscript $nb$ denotes a neighbour point, then at the visited grid point is calculated from

$$T_{P=}\frac{\sum a_{nb}T_{nb}{}^{*}+b}{a_P}$$

Where $T_{nb}{}^{*}$ stands for the neighbour-point value present in the computer storage. For neighbours that have already been visited during the current iteration, is the recently calculated value; for yet-to-be visited neighbours is the value from the previous iteration. In any case, is the latest available value for the neighbour-point temperature. When all grid points have been visited in this way, one iteration of the GS is complete.

## 2.5 BOUNDARY CONDITIONS

Boundary conditions (b.c.) are constraints necessary for the solution of a boundary value problem. A boundary value problem is a differential equation (or system of differential equations) to be solved in a domain on whose boundary a set of conditions is known[6]. CFD and HT problems fall into this category.

There are two major types of boundary conditions that are imposed on the boundary of the computational domain;

Dirichlet boundary condition: Named after Peter Gustav Lejeune Dirichlet, this condition specifies the value that the unknown function needs to take on along the boundary of the domain. In other words, the value at the boundary are known, and so, no extra calculation or iteration is required. For a dependent variable ∅, Dirichlet boundary condition could be summarized thus;

$$\emptyset = f(\emptyset)$$

Where $f$ is a scalar function of ∅ within the domain.

Neumman boundary condition: This type of boundary condition was named after Carl Neumann. When imposed on a differential equation, it specifies that the value that the derivative of a solution is going to take on the boundary of the domain. This means that extra calculation is needed to determine the boundary condition, and, in some cases, it is iterated to until the right boundary value is reached. For a dependent variable ∅, Neumann boundary condition could be summarized thus;

$$\frac{\partial \emptyset}{\partial n} = \hat{n}\nabla\emptyset = f(\emptyset)$$

Where $f$ is a scalar function of ∅ within the domain.

## 2.6 CONVERGENCE CRITERIA

The solution of linear equations by iterative methods requires for convergence that the absolute magnitudes of all the eigenvalues of the iteration matrix should be less than unity[7]. Convergence is the measure of closeness of a value at a point from one iteration to the other. The Gauss-Seidel method converges if the number of roots inside the unit circle is equal to the order of the iteration matrix.

The check for convergence can be expressed thus

$$\varepsilon = \left| \frac{{x_i}^k - {x_i}^{k-1}}{{x_i}^k} \right|$$

For all *i* where *k* is the current iteration and *k-1* is the previous iteration. The aim is to get a value of ε that is as close to zero as possible. However, the impossibility of the iterative methods to attain exactness gives rise to the imposition of a near zero condition. In most cases, the lower the criterion, the more accurate the solution and the more expensive.

## 2.7 DIMENSIONAL ANALSYIS

The equations; momentum, energy and mass conservation equations which constitute the Navier-Stoke equation are statements of natural laws and they must remain valid irrespective of the units employed. It is still possible to solve the equations if we remove the all the units, making them non-dimensional. The dimensionless form of mass, momentum and energy equations is presented below, and the related dimensionless numbers are introduced.

Reference values;

Characteristic length: $L$

$$\text{Reference velocity: } v_0 = \begin{cases} v_\infty \\ \mu/(\rho L) \\ \lambda/(\rho c_p L) \end{cases}$$

Reference temperature difference $\tilde{T} = \Delta T$

Dimensionless variables

| | |
|---|---|
| $\tilde{x} = \frac{x}{L},\ \tilde{y} = \frac{y}{L}$ | Position |
| $\tilde{v} = \frac{v}{v_0}\quad \tilde{u} = \frac{u}{v_0}$ | Velocity |
| $\theta = T/\tilde{T}$ | Temperature |
| $\tau = tL/v_0$ | Time |

$$P = p/(\rho(v_0)^2) \qquad \text{Pressure}$$

Given the momentum equation

$$\frac{\partial u_i}{\partial t} + u_j \frac{\partial u_i}{\partial x_i} = -\frac{1}{\rho}\frac{\partial p}{\partial x_i} + \upsilon \frac{\partial^2 u_i}{\partial x_i x_j}$$

Taking the X-direction and applying the dimensionless variables above, the equation becomes

$$\frac{\partial v_o \tilde{u}}{\partial\, \tau L/v_0} + v_o\tilde{u}\frac{\partial v_o \tilde{u}}{\partial L\tilde{x}} = -\frac{1}{\rho}\frac{\partial P \rho {v_o}^2}{\partial L\tilde{x}} + \upsilon \frac{\partial^2 v_o \tilde{u}}{\partial L^2 \tilde{x}^2}$$

Dividing through by $\frac{{v_o}^2}{L}$ we obtain

$$\frac{\partial \tilde{u}}{\partial \tau} + \tilde{u}\frac{\partial \tilde{u}}{\partial \tilde{x}} = -\frac{\partial P}{\partial \tilde{x}} + \frac{\upsilon}{v_o L}\frac{\partial^2 \tilde{u}}{\partial \tilde{x}^2}$$

If the reference velocity $v_o = \mu/(\rho L)$ , the term $\frac{\upsilon}{v_o L}$ is the ratio of viscous force to inertial force which is simply $\frac{1}{Re}$, the Reynolds number.

If the reference velocity $v_o = \lambda/\left(\rho c_p L\right)$, the term $\frac{\upsilon}{v_o L}$ is the ratio of momentum diffusion to heat which is simply Pr, the Prandtl number.

For the energy equation, the following is obtained

$$\frac{\partial \theta}{\partial \tau} + \tilde{u}\frac{\partial \theta}{\partial \tilde{x}} = \frac{1}{\mathrm{Pe}}\frac{\partial^2 \theta}{\partial \tilde{x}^2}$$

Where $\mathrm{Pe} = \mathrm{Re} * \mathrm{Pr}$

The above formulation is for forced convection. In the case of natural (free) convection, the momentum equation is given by

$$\frac{\partial \tilde{u}}{\partial \tau} + \tilde{u}\frac{\partial \tilde{u}}{\partial \tilde{x}} = -\frac{\partial P}{\partial \tilde{x}} + Pr\frac{\partial^2 \tilde{u}}{\partial \tilde{x}^2} - Ra * Pr(\theta - \theta_o)$$

Where $Ra = Gr * Pr$ ,Rayleigh Number

$Gr = \frac{\beta g L^3 \Delta T}{(\mu/\rho)^2}$ , Grashoff Number which is the ratio of buoyancy and viscous forces

$\beta$ is the thermal expansion coefficient, in which according to the Boussinesq model, density and buoyancy forces are linear function of temperature.

$$(\rho_o - \rho) = -\rho\, \beta(T - T_o)$$

## 2.8 TURBULENCE MODELLING

So far, in the presentation of the Navier-Stokes equation in previous sections, it has not been stated whether the flow is laminar or turbulent. While laminar flow is a regime of fluid motion in parallel layers; there is no lateral mixing, no cross currents perpendicular to the direction of flow, nor eddies (or swirls) of fluids, turbulent flow is a regime of fluid motion with "random" and chaotic three-dimensional vorticity[8].

The development of turbulence in a flow depends on the velocity of the flow and the viscosity of the fluid. Turbulence increases with velocity (inertia force) and decreases with viscosity (viscous force). Alternatively, it could be stated that turbulence increases with the ratio of the inertia force and the viscous force. The ratio of these forces is known as the Reynolds number *Re* which is expressed thus;

$$\text{Re} = \frac{\text{inertia force}}{\text{viscous force}}$$

$$Re = \frac{\rho V L}{\mu}$$

where $\rho$ is the density of the fluid, $V$ is the velocity of flow and L the characteristic length and $\mu$ is the dynamic viscosity. Therefore, the Reynolds number is the criteria that determines whether a flow is laminar of turbulent. Above a critical number of *Re,* flow is turbulent, and below the critical number, flow is laminar.

Of the three theories about turbulence, Kolmogorov's "energy cascade" concept [9] is the most accepted. Kolmogorov's theory describes how energy is transferred from larger to smaller eddies; how much energy is contained by eddies of a given size; and how much energy is dissipated by eddies of each size. The smallest eddies are of scales at which the molecular viscosity is very effective at dissipating the turbulent kinetic energy as heat[3]. The smallest turbulent eddies are characterized by the Kolmogorov micro length (η) and time ($t_\eta$) scales given by

$$\eta = \left(\frac{\upsilon^3}{\varepsilon}\right)^{\frac{1}{4}}$$

$$t_\eta = \left(\frac{\upsilon}{\varepsilon}\right)^{\frac{1}{2}}$$

Without heavy cost, it is possible to directly simulate laminar flows using the governing equations. However, turbulent flows constitute a significant challenge. The direct simulation of the governing Navier-Stokes equation (called "DNS") is possible only for

"simple" cases and low Re number. For high Re, there is a need for a very small-time step and fine mesh. This is rather highly costly as it can be shown that the number of mesh points scales as $Re^{9/4}$, while realistic applications have a Re $\sim 10^6 - 10^9$. Due to its prohibitive computational cost the DNS approach cannot currently be employed to solve industrial problems[3].

**Turbulence Models**

The high cost of the DNS for turbulent flows has led to the development of turbulent models used to predict the effects of turbulence. The construction and use of mathematical models employed in this prediction is referred to as turbulence modelling. In general, the approach for solving turbulent flow equations can roughly be divided into four classes as Direct Numerical Simulation (DNS), Reynolds Average Navier-Stokes equation (RANS), Large Eddy Simulation (LES) and Detached Eddy Simulation (DES).

The key concept in the LES is to filter the Navier-Stokes equation to determine which scales to keep and which scales to discard[10]. This feature allows for the explicit solution for the large eddies in a calculation and the implicit account for the small eddies using a subgrid-scale model (SGS model).

The most popular approach for tackling industrial turbulent flow problems is the one based on solving the Reynolds Averaged Navier-Stokes (RANS) equations[3] where the statistical averaging is now based not on spatial averaging but on a proper time. The numerical simulation is driven by a turbulence model which is arbitrarily selected to find out the effect of turbulence fluctuation on the mean fluid flow. The idea is to model all scales of turbulent flow. Therefore, approach removes the limitation posed by the mesh size as in the DNS and LES approaches. RANS-based models can be categorized into eddy viscosity models and Reynolds stress model.

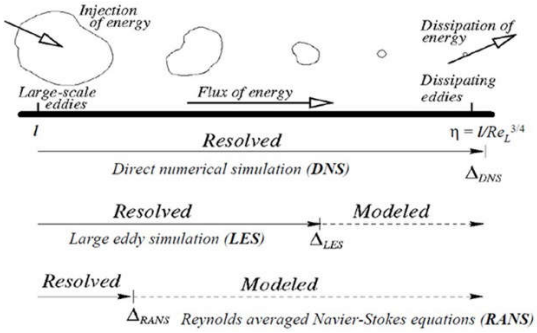



*Figure 4: Turbulence Models*

## CHAPTER THREE

# 3.0 METHODS

## 3.1 PURE DIFFUSION

The heat transfer simulation of the square cross-section of a long rod is carried out. The rod is composed of four materials M1-M4 as shown in the figure 4. The initial temperature field is T = 11.00°C. The problem coordinates, physical properties and the boundary conditions are presented in tables 1, 2 and 3 respectively.

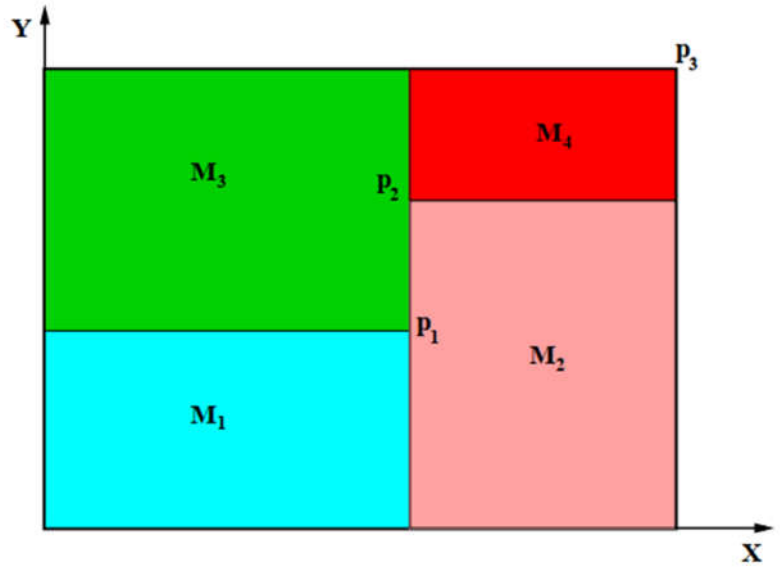



*Figure 5: Schema for the heat conduction problem*

| | x (m) | y (m) |
|---|---|---|
| $p_1$ | 0.30 | 0.40 |
| $p_2$ | 0.30 | 0.80 |
| $p_3$ | 0.90 | 0.90 |

*Table 1: Problem coordinates*

| **Material** | **$\rho$(kg.m$^{-3}$)** | **$c_p$(J.kg$^{-1}$ K$^{-1}$)** | **(W.m$^{-1}$ K$^{-1}$)** |
|---|---|---|---|
| $M_1$ | 2500.0 | 970.0 | 180.0 |
| $M_2$ | 2700.0 | 930.0 | 140.0 |
| $M_3$ | 2200.0 | 710.0 | 150.0 |
| $M_4$ | 1700 | 920.0 | 140.0 |

*Table 2: Physical properties*

| **Boundary** | **Boundary Condition** |
|---|---|
| Bottom | Isotherm at T = 18.00℃ |
| Top | Uniform $Q_{flow} = 89.00$ W/m |
| Left | $T_g = 35.00$℃ $\alpha_g = 8.00$ W/$m^2K$ |
| Right | T(t)= 11.0 + 0.006t℃ |

*Table 3: Boundary conditions*

Following the FVM as presented in [2], the solution to the problem is implemented in a C++ program.

### 3.11 Discretization

The transient 2-dimensional heat conduction problem is governed by the diffusion equation;

$$\rho c_p \frac{\partial T}{\partial t} = \nabla \cdot (k \nabla T)$$

Applying surface integral over a control volume A, the equation becomes

$$\int_S \rho c_p \frac{\partial T}{\partial t} \mathrm{dA} = \int_S \nabla \cdot (k \nabla T) dA$$

Assuming that the material is constant throughout the control volume and applying divergence theorem, we have that;

$$\rho c_p \frac{\partial T}{\partial t} \mathrm{S} = k \oint_C \nabla \mathrm{T} \cdot \hat{n} ds$$

The integral of the RHS is thus calculated

$$\rho \mathrm{c_p} \frac{\partial \mathrm{T}}{\partial \mathrm{t}} \mathrm{S} = \sum_{\text{faces of S}} \mathrm{k_f} \left( \frac{\partial \mathrm{T}}{\partial \mathrm{x}}\bigg|_{\mathrm{f}}, \frac{\partial \mathrm{T}}{\partial \mathrm{y}}\bigg|_{\mathrm{f}} \right) \cdot (\mathrm{n_x}, \mathrm{n_y}) \mathrm{D_f}$$

$\mathrm{D_f}$ is the length of the faces and n being the unit vector in the direction of the faces.

Considering an elemental control volume P with neighbour point E, W, N, S and faces e, w, as shown in the figure below

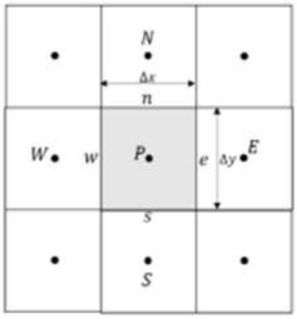



*Figure 6: Control volume*

$\mathrm{D_e} = \mathrm{D_w} = \Delta y \qquad \mathrm{D_n} = \mathrm{D_s} = \Delta x$

$\hat{n}_e = (1,0) \qquad \hat{n}_w = (-1,0) \qquad \hat{n}_n = (0,1) \qquad \hat{n}_s = (0,-1)$

For the north face, we obtain

$$k_N\left(\left.\frac{\partial T}{\partial x}\right|_N, \frac{T_N - T_P}{\delta y}\right)\cdot(0,1)\Delta x$$

This is analogous for other faces. And taking the sum of the four faces, it follows that

$$\rho c_p \frac{\partial T}{\partial t} S = k_N \frac{T_N - T_P}{\delta y}\Delta x + k_S \frac{T_S - T_P}{\delta y}\Delta x + k_E \frac{T_E - T_P}{\delta x}\Delta y + k_W \frac{T_W - T_P}{\delta x}\Delta y$$

Applying implicit scheme of temporal discretization as explained in chapter two, the equation becomes

$$\rho c_p \frac{{T_P}^{n+1} - {T_P}^{n}}{\Delta t} S = k_N \frac{{T_N}^{n+1} - {T_P}^{n+1}}{\delta y}\Delta x + k_S \frac{{T_S}^{n+1} - {T_P}^{n+1}}{\delta y}\Delta x + k_E \frac{{T_E}^{n+1} - {T_P}^{n+1}}{\delta x}\Delta y + k_W \frac{{T_W}^{n+1} - {T_P}^{n+1}}{\delta x}\Delta y$$

In a simplified way, the discretized equation has the final form;

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

The coefficients $a_P$, $a_N$, $a_S$, $a_E$, $a_W$ and $b$ are defined as follows;

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

$$a_N = \frac{k_N \Delta x}{\delta y};$$

$$a_S = \frac{k_S \Delta x}{\delta y};$$

$$a_E = \frac{k_E \Delta y}{\delta x};$$

$$a_W = \frac{k_W \Delta y}{\delta x};$$

$$b = \frac{\rho c_p S {T_P}^{n}}{\Delta t};$$

$$S = \Delta x \Delta y$$

**Material Interface**

Due to the composite nature of material i.e. the non-homogeneity of the material, there exists a possibility of arising non-uniform conductivity between the materials. So, assuming a linear variation of conductivity between two neighbouring grid points fig , we can overcome the non-uniform conductivity by modifying the definition of the coefficient such that if the face of a point lies at the interface of two or more materials, the harmonic mean of the conductivities of the materials are calculated. This achieved as it is presented by Patankar[2].

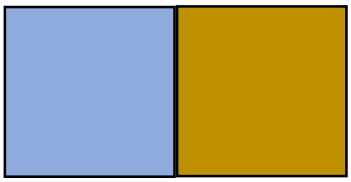


*Figure 7: Material interface*

Therefore, considering the position of the interfaces of the materials and the distance from the center of the control volume the coefficients are given by

control volume the coefficients are given by

$$a_N = \frac{\Delta x}{\frac{y_{int} - y_p}{k_P} + \frac{y_N - y_{int}}{k_N}};$$

$$a_S = \frac{\Delta x}{\frac{y_P - y_{int}}{k_P} + \frac{y_{int} - y_S}{k_S}};$$

$$a_E = \frac{\Delta y}{\frac{x_{int} - x_p}{k_P} + \frac{x_E - x_{int}}{k_E}};$$

$$a_W = \frac{\mathrm{k_W}\Delta y}{\frac{x_P - x_{int}}{k_P} + \frac{x_{int} - x_W}{k_W}};$$

$x_P$ is the position of the CV in the x-direction from the origin

$x_{int}$ is the position of the interface in the x-direction from the origin

$y_P$ is the position of the CV in the y-direction from the origin

$y_{int}$ is the position of the interface in the y-direction from the origin

### 3.12 Boundary Conditions

The boundary conditions as shown in table 3 require additional equations to calculate the temperature at each time step. The equation is given in [2] as follows;

$$a_B T_B = a_I T_I + b$$

Where $a_I$ is the coefficient of the internal neighbour of the boundary point

$T_I$ is the temperature of the internal neighbour point

$b$ is the flux

For the bottom boundary, no additional equation is needed, the nodes at the bottom boundary assume the value of the isotherm, $18.0^0$C.

For the right boundary, it is also straight forward, the temperature at the right side is a function of time.

$$\mathrm{T(t)= 11.0 + 0.006t^{\circ}C}$$

A new value of the right boundary is calculated at every time step.

For the top boundary which is in contact with a uniform $Q_{flow} = 89.00$ W/m, the equation is given by

$$T_{top} = \frac{a_S T_S + b}{a_{top}}$$

$$a_S = \frac{k_S}{\delta y}$$

$$b = 89.00 \text{ W/m}$$

$$a_{top} = a_s$$

For the left boundary which is in contact with a fluid with temperature $35.0^0$C and heat transfer coefficient $\propto_g = 8.00$ W/$m^2k$, the equation is given by

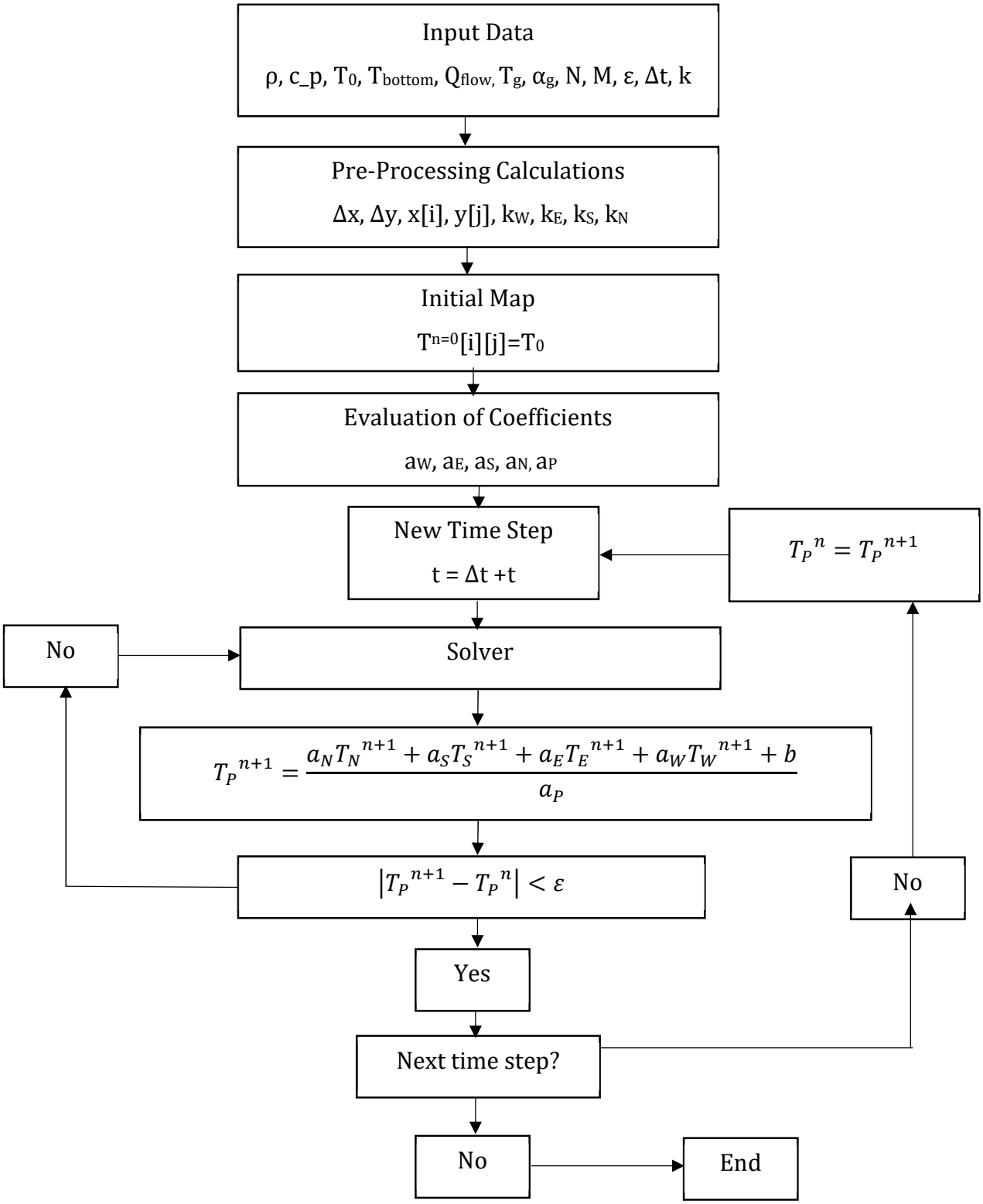
$$T_{left} = \frac{a_E T_E + b}{a_{left}}$$

$$a_E = \frac{k_E}{\delta x}$$

$$b = T_{fluid}{}^* \propto_g$$

$$a_{left} = a_E + \propto_g$$

### 3.13 Algorithm



### 3.14 Results

Simulation of transient heat transfer for a 4-material problem has been performed using the created C++ code. The problem is simulated for a time of 10000s. Utilizing bi-linear interpolation, values of temperature at two different points were obtained. The evolution of the temperature at the given points are presented as follows;

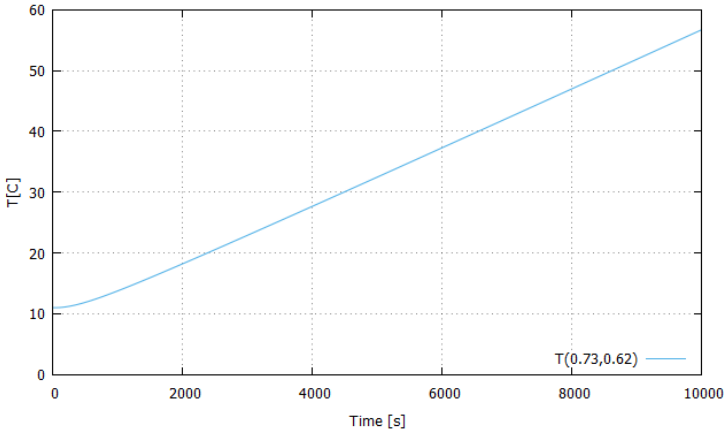



*Figure 8: Temperature evolution at (0.73, 0.62)*

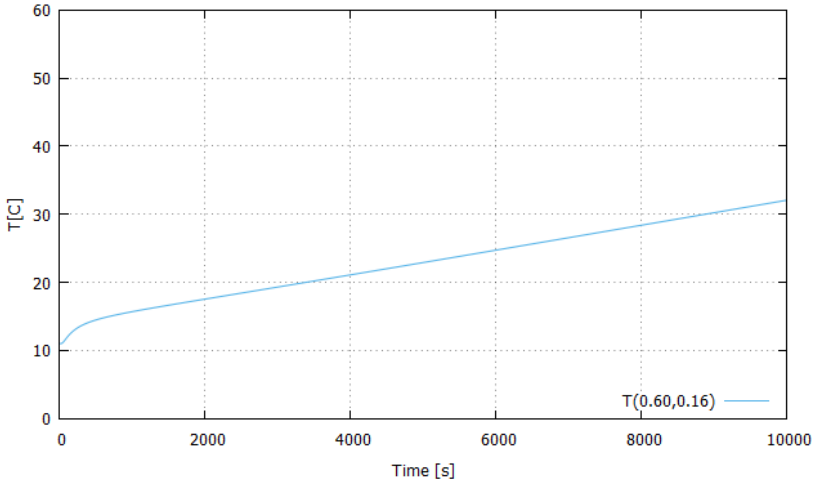



*Figure 9: Temperature evolution at (0.60, 0.16)*

Also, the heat-maps showing temperature distribution on the domain at time t=5000s and t=10000s

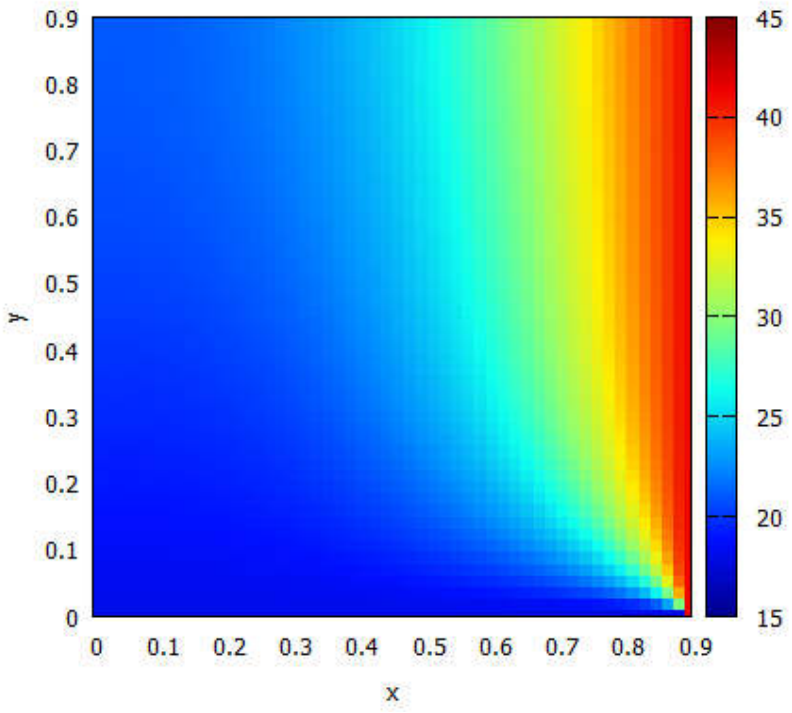



*Figure 10: Temperature Distribution at t=5000s*

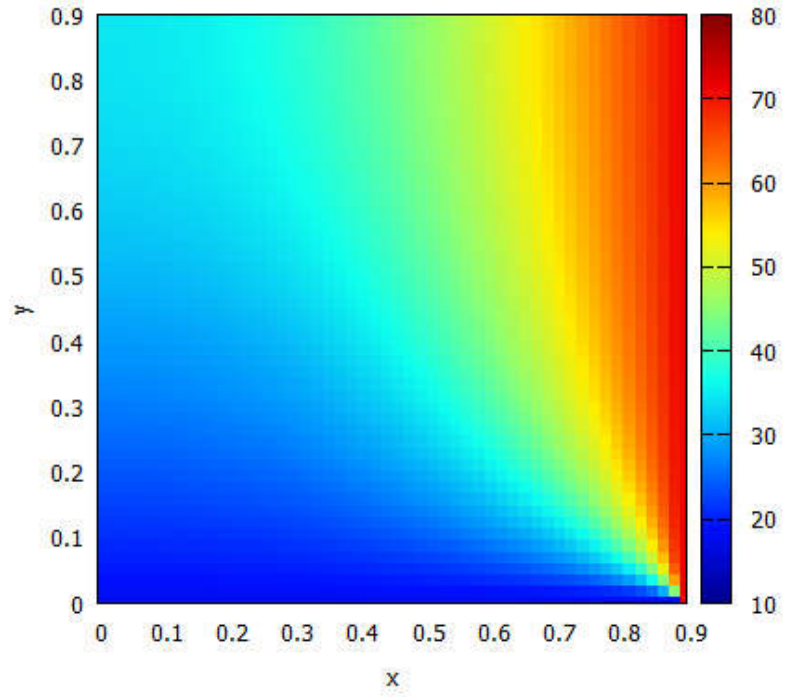



*Figure 11: Temperature Distribution at t=10000s*

### 3.2 CONVECTION-DIFFUSION

The simulation of two distinct cases governed by the convection-diffusion equation is carried out. The equation is written thus;

$$\frac{\partial(\rho\emptyset)}{\partial t} + \nabla \cdot (\rho\emptyset\mathbf{v}) = \nabla \cdot (\Gamma\nabla\emptyset) + S$$

For both cases, the velocity field **v** is given to study the distribution of the general variable $\emptyset$ which is dependent on the diffusion coefficient $\Gamma$. It should be noted that $\emptyset$ can be temperature, enthalpy, concentration, pressure etc.

The analytical solution of these cases is provided as benchmark for the solution obtained in these simulations.

A C++ program based on FVM is created to solve the problems.

**Case 1:** Diagonal Flow

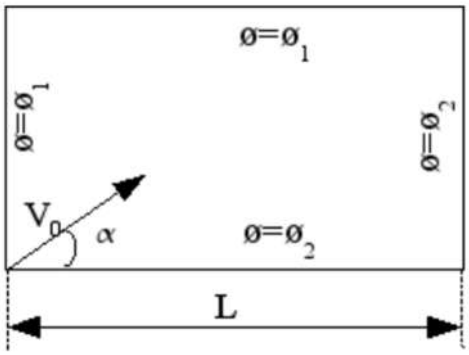



*Figure 12 : Diagonal Flow*

The velocity field is given as follows

$$u(x, y) = V_o \cdot cos(\alpha)$$
$$v(x, y) = V_o \cdot sin(\alpha)$$

The initial condition is given as follows

$\emptyset = \emptyset_1$ above the diagonal

$\emptyset = \emptyset_2$ below the diagonal

**Case 2:** Smith-Hutton Flow

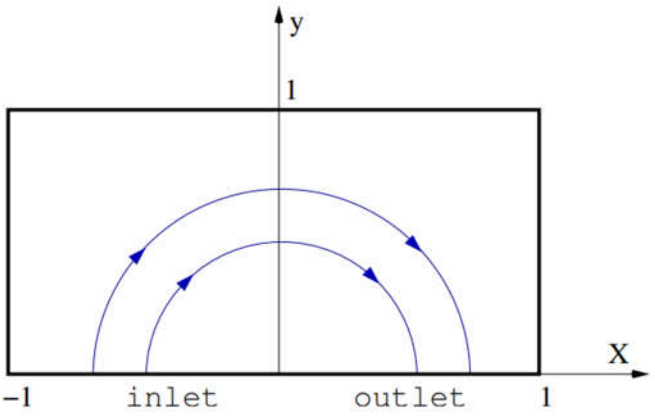



*Figure 13: Smith-Hutton Problem*

The velocity field is given as follows

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

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

Boundary conditions;

$\emptyset = 1 + tanh\big(\alpha(2x+1)\big);$ inlet

$\frac{\partial \emptyset}{\partial y} = 0;$ outlet

$\emptyset = 1 - tanh\,(\alpha);$ elsewhere

$\alpha = 10$

### 3.21 Discretization

As stated earlier, the cases considered are governed by the equation

$$\frac{\partial(\rho\emptyset)}{\partial t} + \nabla \cdot (\rho\emptyset\mathbf{v}) = \nabla \cdot (\Gamma\nabla\emptyset) + S$$

The $S$ term in the equation is dropped because there is neither energy production nor utilization thus, we have

$$\frac{\partial(\rho\emptyset)}{\partial t} + \nabla \cdot (\rho\emptyset\mathbf{v} - \Gamma\nabla\emptyset) = 0$$

The second term of this equation can be described as the total flux (convection plus diffusion) which is defined as follows

$$\vec{J} \equiv \rho\emptyset\mathbf{v} - \Gamma\nabla\emptyset$$

The two-dimensional form of the equation, expressing the gradient, becomes

$$J_x \equiv \rho\emptyset\mathrm{u} - \Gamma\frac{\partial\emptyset}{\partial\mathrm{x}}$$

$$J_y \equiv \rho\emptyset\mathrm{v} - \Gamma\frac{\partial\emptyset}{\partial\mathrm{y}}$$

The Peclet number P, which describes the strength of the convective term with respect to the diffusive term is introduced and defined as follows.

$$P_x = \frac{\rho\mathrm{u}\Delta\mathrm{x}}{\Gamma}$$

$$P_y = \frac{\rho\mathrm{v}\Delta\mathrm{y}}{\Gamma}$$

For higher values of Peclet number, convection is predominant while diffusion is predominant in flows with low values of Peclet number.

Taking the horizontal and vertical velocity components $u$ & $v$ as well as the distance $\Delta x_i$ to be 1, $P_x = P_y$.

The fluxes at the faces of the control volume are obtained by integrating $J_x$ and $J_y$ over the respective faces. For example, $J_e$ is the $\int J_x dy$ over the interface $e$ and so on.

The final discretized equation is given by

$$a_P\emptyset_P = a_N\emptyset_N + a_S\emptyset_S + a_E\emptyset_E + a_W\emptyset_W + b$$

Where $a_N = D_n A(|P|) + max\{-F_n, 0\}$,

$a_S = D_s A(|P|) + max\{F_s, 0\}$,

$a_E = D_e A(|P|) + max\{-F_e, 0\}$,

$a_W = D_w A(|P|) + max\{F_w, 0\}$,

$a_P = a_N + a_S + a_E + a_W + a^0{}_P$

$$a^0{}_P = \frac{\rho^0{}_P \Delta x \Delta y}{\Delta t}$$

$$b = a^0{}_P \emptyset^0{}_P$$

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

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

$\delta y$ and $\delta x$ are the distances between two grid points in the direction of the face considered. It should be noted that notations with upper case such as N, S, E and W represent the grid points while the lower case of the alphabets n, s, e, and w represent the face of the control volume.

**Convective Schemes**

Despite convective term's seeming simplicity, its discretization poses some challenges. Up until now, the piecewise linear profile $\phi$ has been the result of the normal discretization which is otherwise known as the central difference scheme. However, unrealistic results are bound to be obtained when the condition of flow does not favour the use of the scheme[2]. As such researchers have come up with different schemes in order to evaluate the convective term at the face of the control volume. This is the reason for the term $A(|P|)$ in the formulation of the coefficients. The function is based on the scheme being used. The schemes are presented below.

- Central Difference (CDS): It is a second order scheme, variable at the cell face is calculated as an arithmetic mean. That is:

$$\phi_e = 0.5(\phi_E + \phi_C)$$

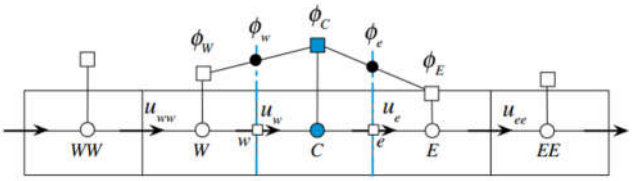



*Figure 14: CDS*

- Upwind Difference Scheme (UDS): It is a first order scheme and the value of $\phi$ at the cell face is equal to the value of $\phi$ at the grid point on the upwind side of the face. That is:

$$\phi_e = \phi_C if F_e > 0$$
$$\phi_e = \phi_E if F_e < 0$$

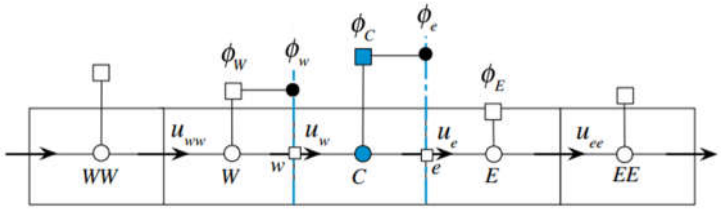



*Figure 15: UDS*

- Hybrid Difference Scheme (HDS): Uses CDS for low velocities and UDS for high velocities.
- Exponential Difference Scheme (EDS): It is a second order scheme and the evaluation of the dependent variable at the cell face comes from the exact

solution of the convection-diffusion equation in one-dimensional, null source term and steady problem.

- Power-law Difference Scheme (PLDS): It is a second order scheme and variable at the cell face is calculated with an approximation of the EDS by a polynomial of fifth degree.

The following table summarizes the value of $A(|P|)$ for the schemes.

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

*Table 4: Value of A(P) for convective schemes*

For improvement of accuracy, it is important to use more than two nodal values to evaluate $\phi$. This has led to the introduction of the high order numerical schemes such as QUICK, SMART, SUDS.

The summary of the formulation of the higher scheme is given in the following table.

| Numerical scheme | Order | $\overline{\phi}_f$ |
|---|---|---|
| UDS | 1 | $\overline{\phi}_f = \overline{\phi}_C$ |
| UDS2 | 2 | $\overline{\phi}_f = \frac{3}{2}\overline{\phi}_C$ |
| QUICK | 3 | $\overline{\phi}_f = \frac{3}{8} + \frac{3}{4}\overline{\phi}_C$ |
| SMART | $2-4$ | $\overline{\phi}_f = 3\overline{\phi}_C$ if $0 < \overline{\phi}_C < \frac{1}{6}$<br>$\overline{\phi}_f = \frac{3}{8} + \frac{3}{4}\overline{\phi}_C$ if $\frac{1}{6} < \overline{\phi}_C < \frac{5}{6}$<br>$\overline{\phi}_f = 1$ if $\frac{5}{6} < \overline{\phi}_C < 1$<br>$\overline{\phi}_f = \overline{\phi}_C$ otherwise |

*Table 5: Higher convective schemes*

Considering the normalized variable profile shown below

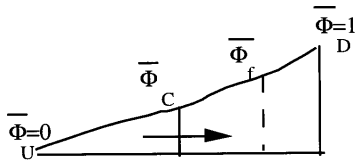



*Figure 16: Normalized variable profile*

$$\acute{\emptyset}_f = \frac{\emptyset_f - \emptyset_U}{\emptyset_D - \emptyset_U}$$

The involvement of the high order scheme leads to the addition of a differed term $b_{de}$to the general equation such that we have that

$$a_P\emptyset_P = a_N\emptyset_N + a_S\emptyset_S + a_E\emptyset_E + a_W\emptyset_W + b + b_{de}$$

Where

$$b_{de} = F_e\left(\phi^{UDS}{}_e - \phi^{HS}{}_e\right) - F_w\left(\phi^{UDS}{}_w - \phi^{HS}{}_w\right) + F_n\left(\phi^{UDS}{}_n - \phi^{HS}{}_n\right) - F_s\left(\phi^{UDS}{}_s - \phi^{HS}{}_s\right)$$

$\phi^{HS}{}_f$ is the variable evaluated at the cell face $f$ with high order numerical scheme.

### 3.22 Boundary Conditions

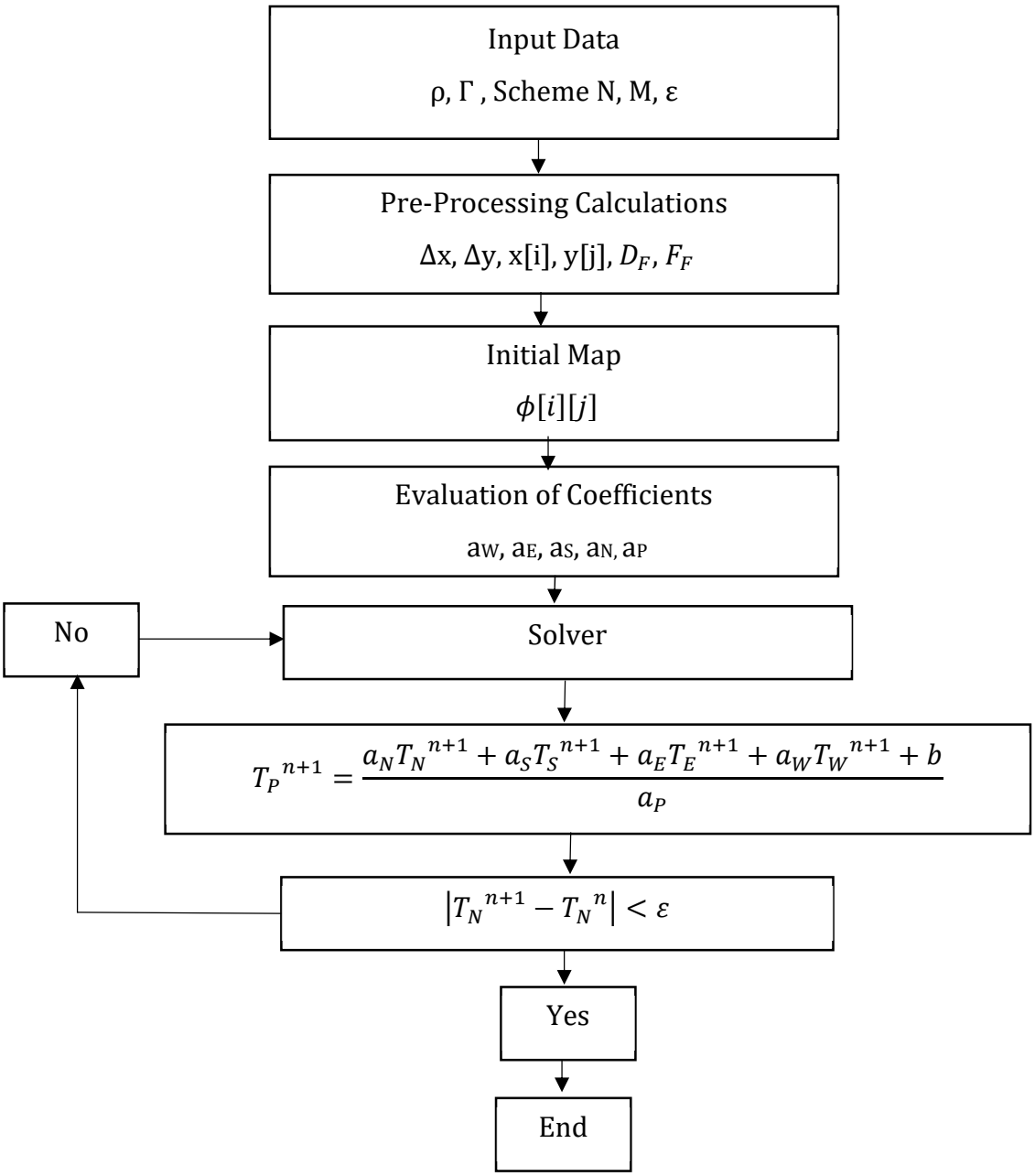
Case 1: A Dirichlet boundary condition prevails throughout this case. The values at the boundary assume the value already given.

$$\emptyset_P = \emptyset_S$$

Case 2: A Dirichlet boundary condition is imposed on all boundary except the outlet boundary where the Neumann boundary condition is used; the boundary values assume the values of the neighbouring internal node in the direction perpendicular to the boundary. The outlet boundary condition is given by $\frac{\partial \emptyset}{\partial y} = 0$;

$$\emptyset_P = \emptyset_N$$

### 3.23 Algorithm



### 3.24 Results

Implementing the created C++ code, two cases governed by the convection-diffusion equation has been simulated. The steady state solution of the problems was obtained. The distribution of ∅ in the computational domain for both cases were obtained. Effects of changes in different parameters were investigated.

**Diagonal Flow**

The distribution for ∅ using UDS and a grid size of $n_x = n_y = 20$ is presented below

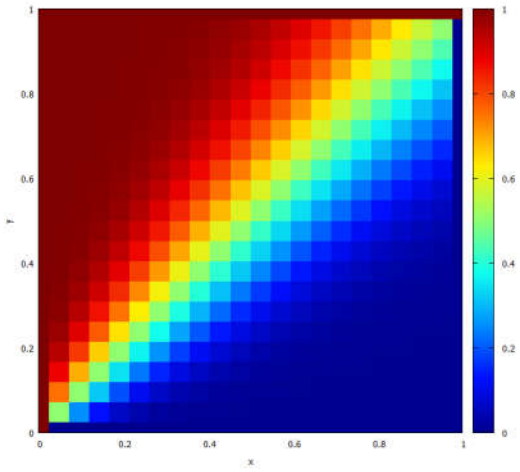


*Figure 17: Distribution of Φ*

It would be noticed from the figure above that there exists a sharp transition in the distribution map. This is as a result of the phenomena known as *False Diffusion*. According to Patankar[2], false diffusion occurs when the flow is oblique to the grid line and when there is a nonzero gradient of the dependent variable in the direction normal to the flow. As in the case being studied, false diffusion is most serious when the flow makes an angle of $45^0$.

This effect seems to lessen with increase in grid size as presented in the following figures;

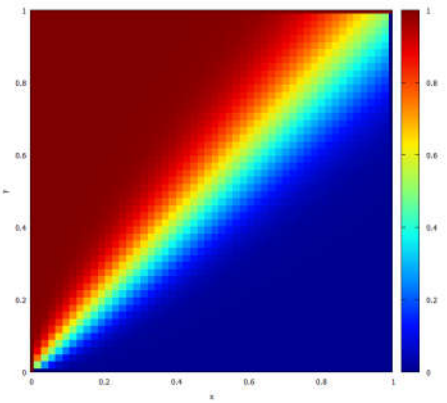


*Figure 18: nx=ny= 50*

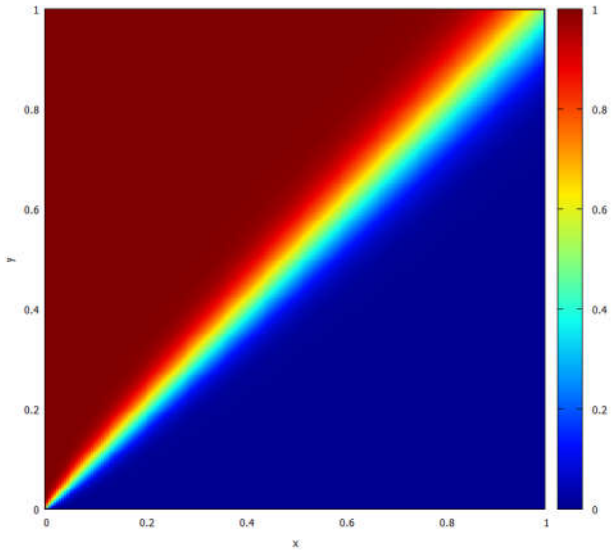



*Figure 19: nx=ny =100*

**Smith-Hutton Problem**

The ∅ distribution for the various values of $\frac{\rho}{\Gamma}$ using UDS and a grid size of $n_x = 2n_y = 100$ are presented below

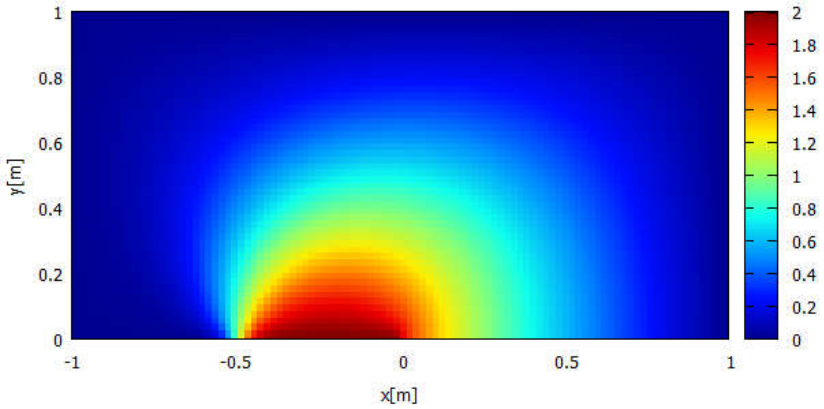



*Figure 20: Distribution for Pe=10*

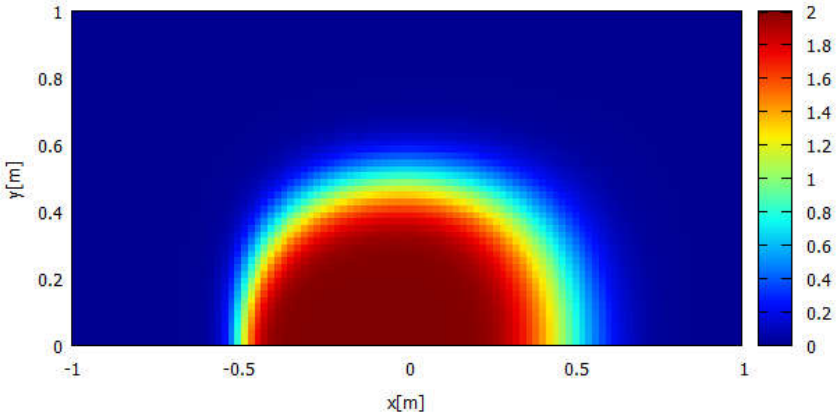



*Figure 21: Distribution for Pe=1000*

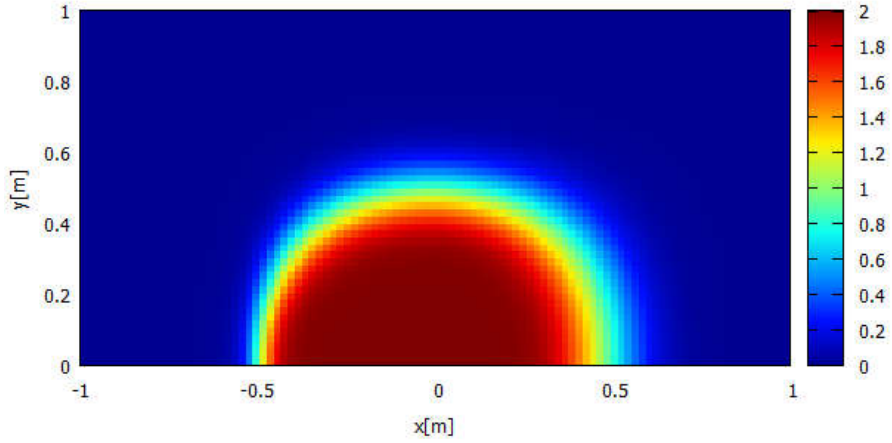



*Figure 22: Distribution for Pe=1e06*

Also, a plot of the values at the outlet of the domain for the values of $\frac{\rho}{\Gamma}$ is presented

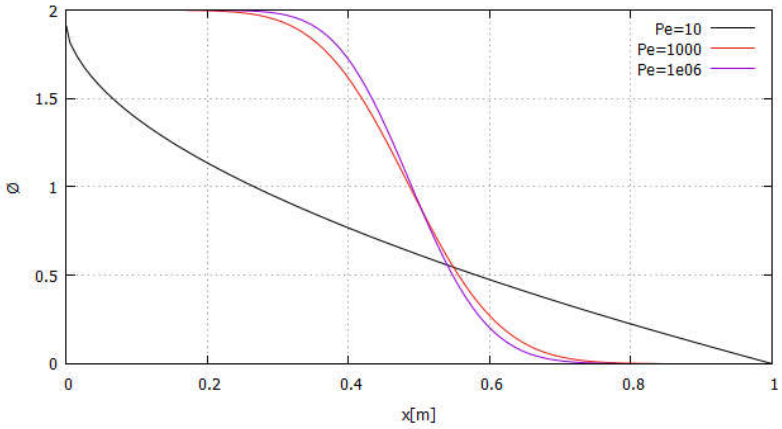



*Figure 23: Outlet distribution of Φ*

**Effect of Mesh Density**

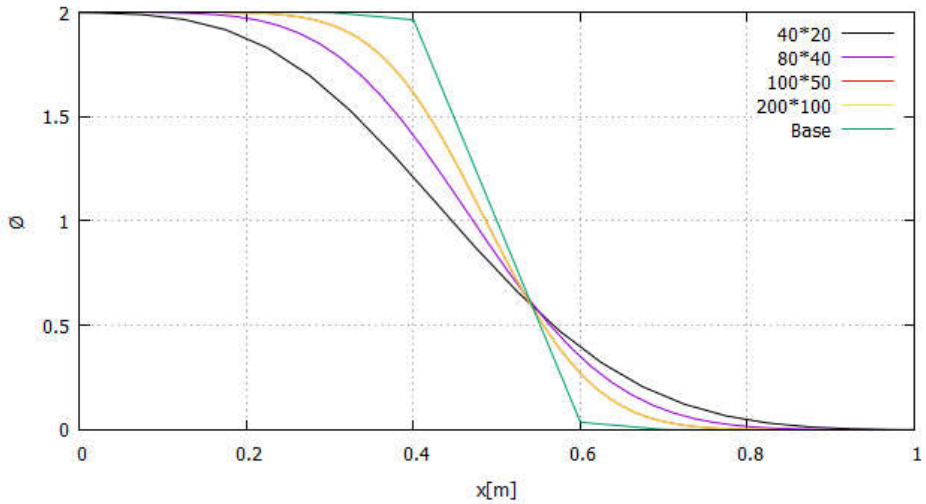



*Figure 24:Mesh effect*

When compared with the base values, simulations at $\frac{\rho}{\Gamma}$=1000 with higher grid densities were closer to the base values than those with coarse densities.

**Effect of Convective Scheme**

Low resolution and high order schemes were also applied for the simulation of $\frac{\rho}{\Gamma}$= {10,1e03,1e06} and grid size $n_x = 2n_y = 200$ are presented as follows

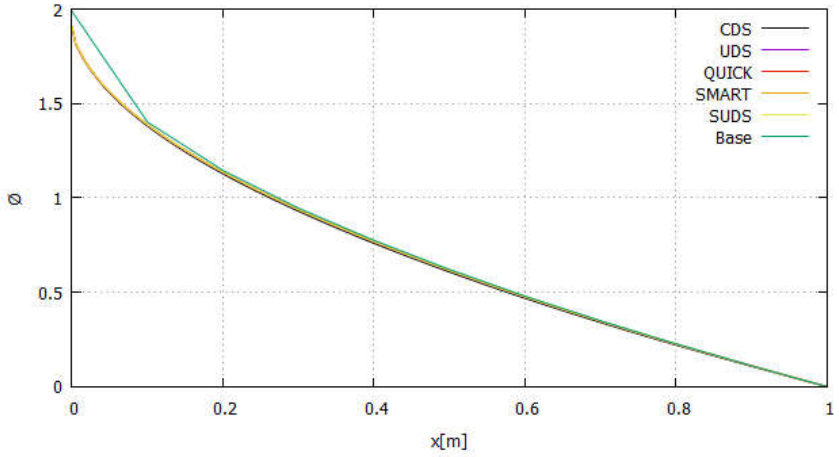



*Figure 25Different schemes Pe=10*

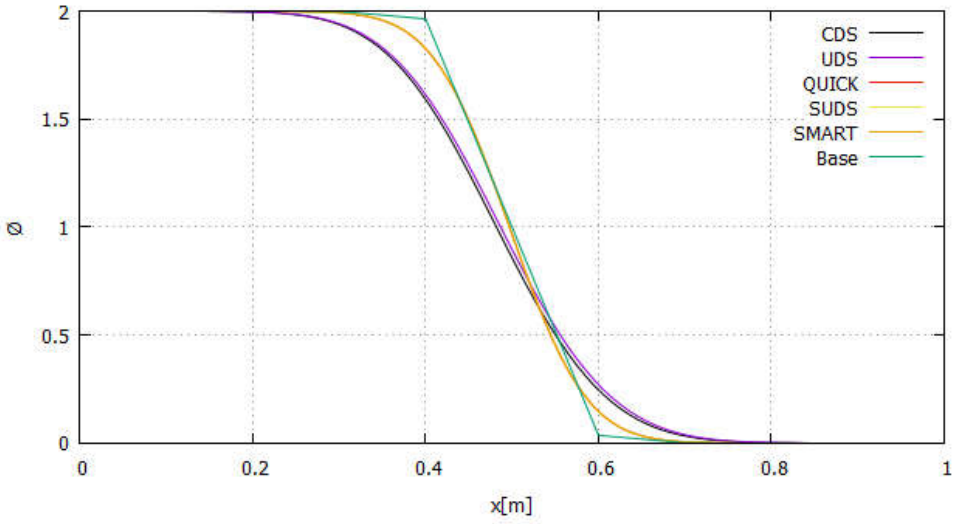



*Figure 26Different Schemes =1000*

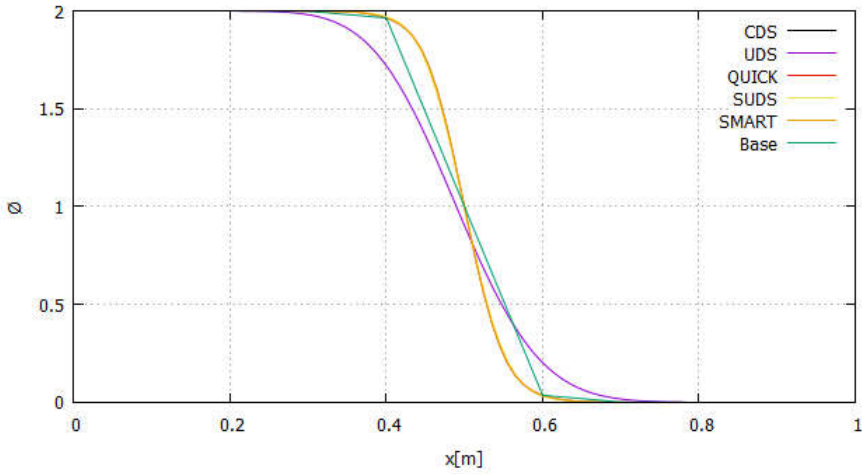



*Figure 27:Different Schemes Pe=1e06*

Generally, it observed that higher resolution schemes like SMART, SUDS, and QUICK offer better accuracy compared to UDS and CDS for the same number of grid points. However, taking a closer look at the curves as in the figures below, the average difference between SMART, SUDS, and QUICK is very low (~1e05).

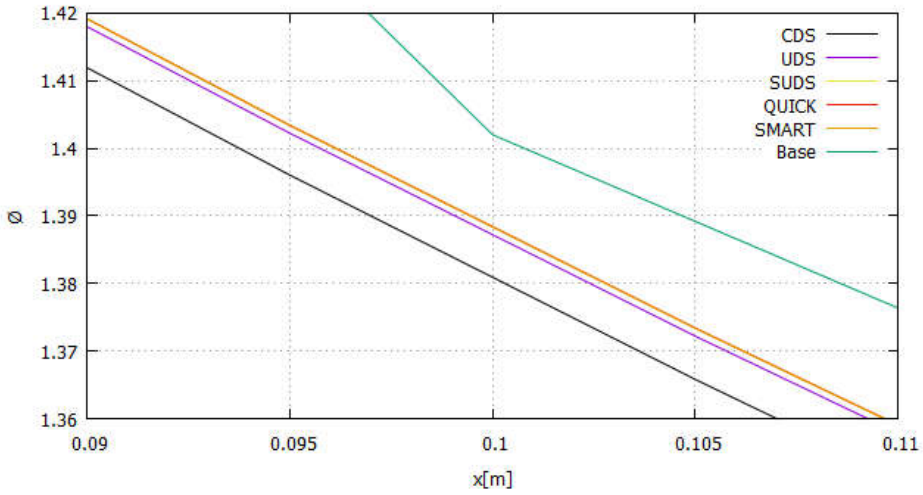



*Figure 28: Different Scheme Pe=10 (Zoomed)*

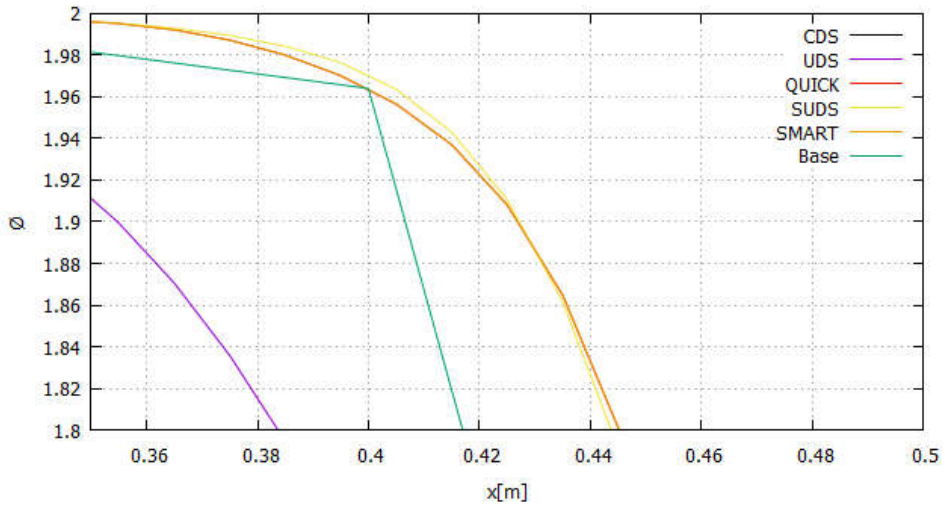



*Figure 29: Different Scheme Pe=1e06 (Zoomed)*

## 3.3 THE DRIVEN CAVITY PROBLEM

The simulation of an incompressible flow within a square cavity as shown in the figure below is presented. It is assumed that velocity is constant throughout the boundaries of the cavity.

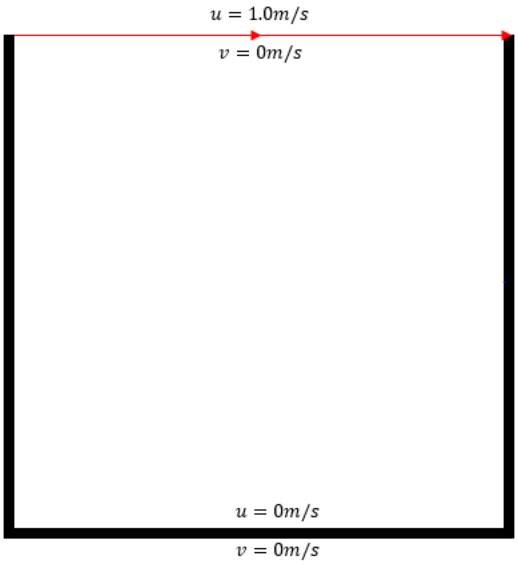



*Figure 30: Driven cavity problem*

A C++ program based on FVM is created to solve the problem with a wide range of Reynolds number.

### 3.31 Fractional Step Method

The Fractional step method (FSM), also referred to as the prediction-projection method is a popular method for solving unsteady incompressible Navier-Stokes equation[11].

Given the Navier-Stokes equation

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

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

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

It is quite difficult to solve the equation because of several reasons;

- There is an equation to for **v** but no equation to solve the pressure term $\nabla p$
- The computed velocity fields of **v** from the momentum equation must satisfy the continuity equation
- Non-linearity of the convective term in the momentum equation.
- It is impossible to use equation of state to compute pressure, as the density and temperature might be constant

In order to ensure the incompressibility of the flow the pressure gradient term $\nabla p$ is viewed as a projector into a divergence-free space. This is achieved by the Helmholtz-Hodge decomposition theorem (explained below). And this is the theoretical basis of the FSM.

Introducing a projector operator $\prod(\cdot)$ that projects any vector field onto a divergence-free space.

$$\nabla\cdot\prod(\boldsymbol{a}) = 0$$

The operator has two properties

- $\prod(\nabla\emptyset) = 0$
- If $\nabla\cdot\boldsymbol{a} = 0$, then $\prod(\boldsymbol{a}) = \mathbf{a}$

Applying it to a rearranged Navier stokes equation, we obtain the following equation;

$$\prod\left(\frac{\partial\mathbf{v}}{\partial t} + \nabla p\right) = \prod\left(-(\mathbf{v}\cdot\nabla)\mathbf{v} + \frac{1}{Re}\Delta\mathbf{v}\right)$$

Given the properties of the operator we have that,

$$\prod\left(\frac{\partial\mathbf{v}}{\partial t}\right) = \frac{\partial\mathbf{v}}{\partial t}$$

$$\prod(\nabla p) = 0$$

This implies that the incompressible transient term remains unchanged and the pressure gradient vanishes. Therefore, we have the Navier-stokes equation split into two parts;

$$\frac{\partial \mathbf{v}}{\partial t} = \prod\left(-(\mathbf{v}\cdot\nabla)\mathbf{v} + \frac{1}{Re}\Delta\mathbf{v}\right)$$

$$\nabla p = -(\mathbf{v}\cdot\nabla)\mathbf{v} + \frac{1}{Re}\Delta\mathbf{v} - \prod\left(-(\mathbf{v}\cdot\nabla)\mathbf{v} + \frac{1}{Re}\Delta\mathbf{v}\right)$$

Applying the divergence operator to the equation above, we obtain the following Poisson equation for pressure;

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

Thus, the momentum equation becomes;

$$\frac{\partial \mathbf{v}}{\partial t} = \boldsymbol{R}(\mathbf{v}) - \nabla p$$

Where, $\boldsymbol{R}(\mathbf{v}) \equiv -(\mathbf{v}\cdot\nabla)\mathbf{v} + \frac{1}{Re}\Delta\mathbf{v}$

According to[11] the fractional step method can be summarized thus;

- Evaluation of $\mathbf{R}(\mathbf{v})$
- Evaluation of predictor velocity predictor $\mathbf{v}^p = \mathbf{v}^n + \Delta t\left(\frac{3}{2}\mathbf{R}(\mathbf{v}^n) - \frac{1}{2}\mathbf{R}(\mathbf{v}^{n-1})\right)$
- Evaluation of $\nabla\cdot\mathbf{v}^p$ and solve the Poisson equation $\Delta\tilde{p} = \nabla\cdot\mathbf{v}^p$ , this gives the distribution of the Pseudo pressure $\tilde{p}$
- Velocity of the next instant time $\mathbf{v}^{n+1} = \mathbf{v}^p - \Delta\tilde{p}$

### 3.32 Staggered Meshes

In the evaluation of the fourth step of the FSM, it is important that the pressure gradient at a node is dependent on Pressure of the node. However, utilizing a mesh where both velocity and pressure are calculated at the same node will make the pressure gradient independent of the node pressure. This will consequently give rise to unrealistic result. This is known as the checkerboard problem[2].

This problem could be solved using the staggered mesh as shown in figure 30. This involves the addition of two meshes, where each component of vector field is stored in its related staggered mesh. The mesh arrangement is made such that the center of the x-component node lies on the vertical face of the standard mesh while the center of the y-

component lies on the horizontal face of the standard mesh. The standard mesh stores the scalar field.

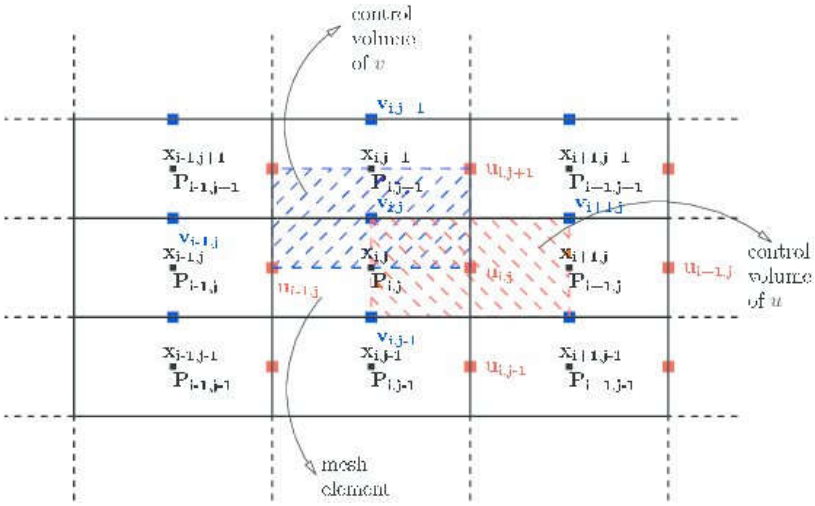



*Figure 31: Staggered Mesh*

### 3.33 Discretization

**Evaluation of $R(\mathbf{v})$**

Recall that $\boldsymbol{R}(\mathbf{v}) = -(\mathbf{v}\cdot\nabla)\mathbf{v} + \frac{1}{Re}\Delta\mathbf{v}$

Resolving the velocity into x and y components;

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

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

Therefore, applying the gauss theorem and integrating $R(u)$ over a CV in the x-mesh, we have that

$$\int_{\mathrm{v}} \boldsymbol{R}(u)\mathrm{dV} = \int_{\mathrm{V}} \left(-(v\cdot\nabla)u + \frac{1}{Re}\Delta u\right)\mathrm{dV}$$

$$-\int_{\mathrm{V}} \nabla\cdot(\mathbf{v}u)\mathrm{dV} + \frac{1}{Re}\int_{\mathrm{V}} \nabla\cdot(\nabla u)\mathrm{dV}$$

$$-\oint_{dv} u\mathbf{v}\cdot\hat{n}dS + \frac{1}{Re}\oint_{dv} \nabla u\cdot\hat{n}dS$$

Approximating the surface integral across the four faces east, west, north and south, we obtain

$$\int_{\mathrm{V}} \boldsymbol{R}(u)\,\mathrm{d}V = \left(-u_e(u_e,v_e)\begin{pmatrix}n_{xe}\\ n_{ye}\end{pmatrix}S_e + \frac{1}{Re}\left(\frac{\partial u}{\partial x}\Big|_e, \frac{\partial u}{\partial y}\Big|_e\right)\begin{pmatrix}n_{xe}\\ n_{ye}\end{pmatrix}S_e\right)$$
$$+\left(-u_w(u_w,v_w)\begin{pmatrix}n_{xw}\\ n_{yw}\end{pmatrix}S_w + \frac{1}{Re}\left(\frac{\partial u}{\partial x}\Big|_w, \frac{\partial u}{\partial y}\Big|_w\right)\begin{pmatrix}n_{xw}\\ n_{yw}\end{pmatrix}S_w\right)$$
$$+\left(-u_n(u_n,v_n)\begin{pmatrix}n_{xn}\\ n_{yn}\end{pmatrix}S_n + \frac{1}{Re}\left(\frac{\partial u}{\partial x}\Big|_n, \frac{\partial u}{\partial y}\Big|_n\right)\begin{pmatrix}n_{xn}\\ n_{yn}\end{pmatrix}S_n\right)$$
$$+\left(-u_s(u_s,v_s)\begin{pmatrix}n_{xs}\\ n_{ys}\end{pmatrix}S_s + \frac{1}{Re}\left(\frac{\partial u}{\partial x}\Big|_s, \frac{\partial u}{\partial y}\Big|_s\right)\begin{pmatrix}n_{xs}\\ n_{ys}\end{pmatrix}S_s\right)$$

Noting that

$S_e = S_w = \Delta y, S_n = S_s = \Delta x, \begin{pmatrix}n_{xe}\\ n_{ye}\end{pmatrix} = (1,0), \begin{pmatrix}n_{xw}\\ n_{yw}\end{pmatrix} = (-1,0), \begin{pmatrix}n_{xn}\\ n_{yn}\end{pmatrix} = (0,1),$

$\begin{pmatrix}n_{xs}\\ n_{ys}\end{pmatrix} = (0,-1),$

$$\boldsymbol{R}(u)dxdy = -u_e^* u_e\Delta y + \frac{1}{Re}\frac{u_E - u_P}{\delta x}\Delta y + u_w^* u_w\Delta y + \frac{1}{Re}\frac{u_W - u_P}{\delta x}\Delta y$$
$$-u_n^* v_n\Delta x + \frac{1}{Re}\frac{u_N - u_P}{\delta y}\Delta x + u_s^* v_s\Delta x + \frac{1}{Re}\frac{u_S - u_P}{\delta y}\Delta x$$

and

$$\boldsymbol{R}(v)dxdy = -v_e^* u_e \Delta y + \frac{1}{Re}\frac{v_E - v_P}{\delta x}\Delta y + v_w^* u_w \Delta y + \frac{1}{Re}\frac{v_W - v_P}{\delta x}\Delta y$$
$$- v_n^* v_n \Delta x + \frac{1}{Re}\frac{v_N - u_P}{\delta y}\Delta x + v_s^* v_s \Delta x + \frac{1}{Re}\frac{v_S - v_P}{\delta y}\Delta x$$

Noting that $u_f \; and \; v_f$ are the velocities of the corresponding face of the control volume obtained by simple interpolation while $u_f^* \; and \; v_f^*$ are the same velocities obtained by convective schemes. $u_F \; and \; u_F$ are the velocities at the neighbour points of the control point in the perpendicular to the faces.

**Evaluation of $\mathbf{v}^p$**

Given by the equation;

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

It can be expressed in the x-component and y-component as follows;

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

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

**Pressure Equation**

Recall that

$$\Delta\tilde{p} = \nabla \cdot \mathbf{v}^p$$

Integrating across a control volume, the equation becomes

$$\int_{\mathrm{V}} \nabla \cdot \nabla\tilde{p} dV = \int_{\mathrm{V}} \nabla \cdot \mathbf{v}^p dV$$

Transforming it to a surface integral using the divergence theorem, we have that;

$$\oint_{dv} \nabla\tilde{p} \cdot \hat{n} dS = \oint_{dv} v^p \cdot \hat{n} dS$$

Approximating the surface integral across the four faces east, west, north and south, we obtain

$$\left(\left(\frac{\partial\tilde{p}}{\partial x}\Big|_e, \frac{\partial\tilde{p}}{\partial y}\Big|_e\right)\binom{n_{xe}}{n_{ye}}S_e\right) + \left(\left(\frac{\partial\tilde{p}}{\partial x}\Big|_w, \frac{\partial\tilde{p}}{\partial y}\Big|_w\right)\binom{n_{xw}}{n_{yw}}S_w\right)$$
$$+\left(\left(\frac{\partial\tilde{p}}{\partial x}\Big|_n, \frac{\partial\tilde{p}}{\partial y}\Big|_n\right)\binom{n_{xn}}{n_{yn}}S_n\right) + \left(\left(\frac{\partial\tilde{p}}{\partial x}\Big|_s, \frac{\partial\tilde{p}}{\partial y}\Big|_s\right)\binom{n_{xs}}{n_{ys}}S_s\right)$$
$$= \left((u^p{}_e, v^p{}_e)\binom{n_{xe}}{n_{ye}}S_e\right) + \left((u^p{}_w, v^p{}_w)\binom{n_{xw}}{n_{yw}}S_w\right)$$
$$+\left((u^p{}_n, v^p{}_n)\binom{n_{xn}}{n_{yn}}S_n\right) + \left((u^p{}_s, v^p{}_s)\binom{n_{xs}}{n_{ys}}S_s\right)$$

Simplification yields

$$\frac{\tilde{p}_E - \tilde{p}_P}{\delta_{x_E}}\Delta y + \frac{\tilde{p}_W - \tilde{p}_P}{\delta_{x_W}}\Delta y + \frac{\tilde{p}_N - \tilde{p}_P}{\delta_{y_N}}\Delta x + \frac{\tilde{p}_S - \tilde{p}_P}{\delta_{y_S}}\Delta x = u_E^p\Delta y - u_W^p\Delta y + v_N^p\Delta x - v_S^p\Delta x$$

The discretized pseudo-pressure equation is thus given as

$$a_P\tilde{p}_P = a_N\tilde{p}_N + a_S\tilde{p}_S + a_E\tilde{p}_E + a_W\tilde{p}_W + b$$

The coefficients $a_P$, $a_N$, $a_S$, $a_E$, $a_W$ and $b$ are defined as follows;

$$a_P = a_N + a_S + a_E + a_W;$$
$$a_N = \frac{\Delta x}{\delta y};$$
$$a_S = \frac{\Delta x}{\delta y}$$
$$a_E = \frac{\Delta y}{\delta x};$$
$$a_W = \frac{\Delta y}{\delta x};$$
$$b = \Delta y\left(-u_E^p + u_W^p\right) + \Delta x\left(-u_N^p + u_S^p\right)$$

**Velocity at next time-Step**

Given by the equation;

$$\mathbf{v}^{n+1} = \mathbf{v}^p - \nabla\tilde{p}$$

It is expressed in the x-component and y-component as follows;

$$u^{n+1} = u^p - \frac{\partial\tilde{p}}{\partial x}$$
$$v^{n+1} = v^p - \frac{\partial\tilde{p}}{\partial y}$$

Note that $\tilde{p} = p\Delta t$

**Timestep**

As a result of stability and convergence issues, the CFL condition was imposed in the selection of a suitable time step. It is given thus;

$$t_{step-c} = min\left(0.35\frac{\Delta x}{|v|}\right)$$

$$t_{step-d} = min\left(0.20\frac{\rho\Delta x^2}{\mu}\right)$$

$$t_{step} = min\left(t_{step-c}, t_{step-d}\right)$$

### 3.34 Boundary Conditions

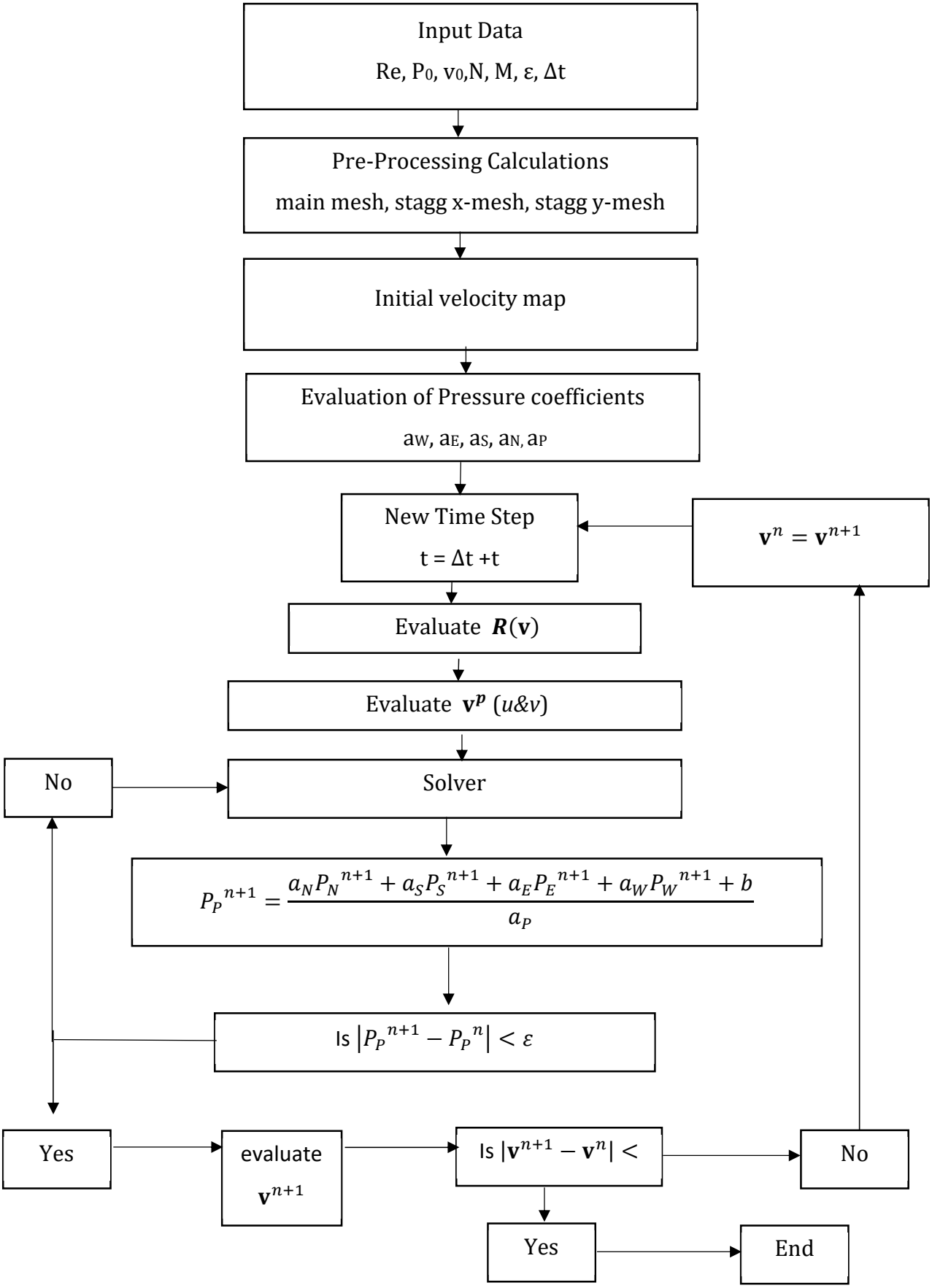
The boundary conditions of the cavity are summarized thus;

| **Boundary** | $u$ | $v$ |
|---|---|---|
| Top | $u_0$ | 0 |
| Bottom | 0 | 0 |
| Right | 0 | 0 |
| Left | 0 | 0 |

*Table 6: Boundary condition (Driven Cavity)*

For the pressure equation, since it was given that $\partial p/\partial n$=0, we have that the pressure at the boundary is equal to the pressure of the neighbour node in the direction perpendicular to the boundary (Neumann Boundary condition). For example, for any point [i][j], lying on the right boundary, we have that p[i][j] = p[i-1] [j]. And it is extended to other boundaries.

### 3.35 Algorithm



### 3.36 Results

Simulation for the driven cavity problem has been carried out using the Fractional step method. Results for Re = 100 and 1000 using a grid size of $n_x = n_y = 100$ are presented as follows;

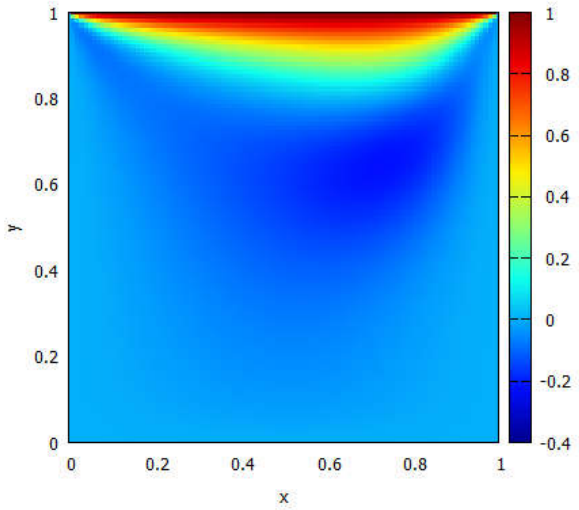



*Figure 32: x Velocity distribution Re=100*

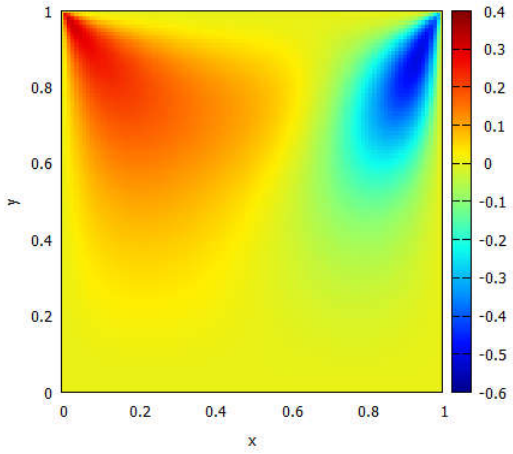



*Figure 33: y Velocity distribution Re=100*

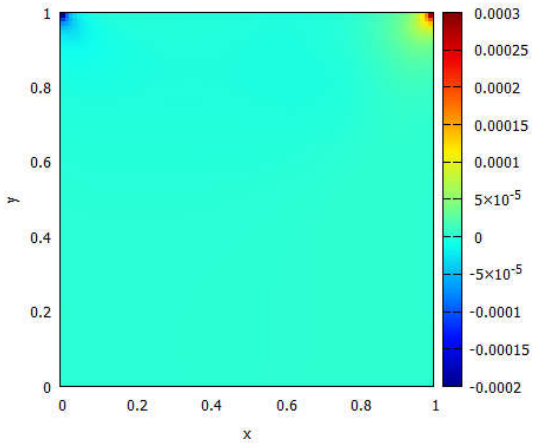



*Figure 34: Pressure distribution Re=100*

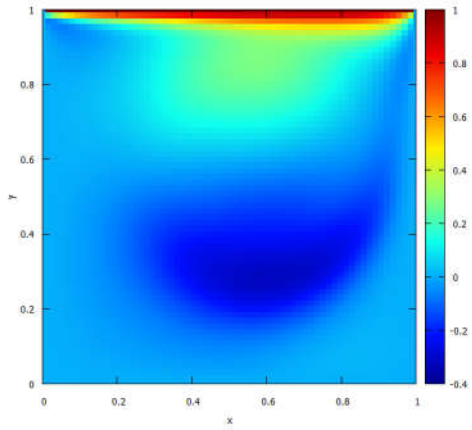



*Figure 35: x Velocity distribution Re=1000*

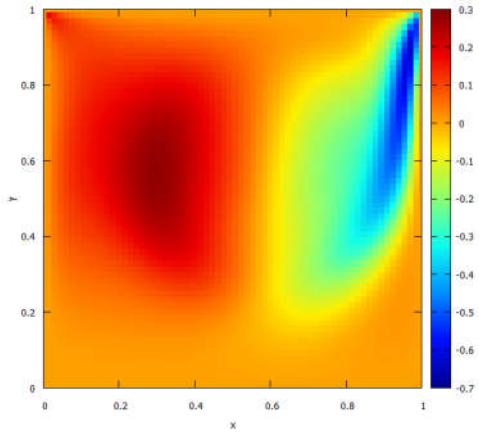



*Figure 36: y Velocity distribution Re=1000*

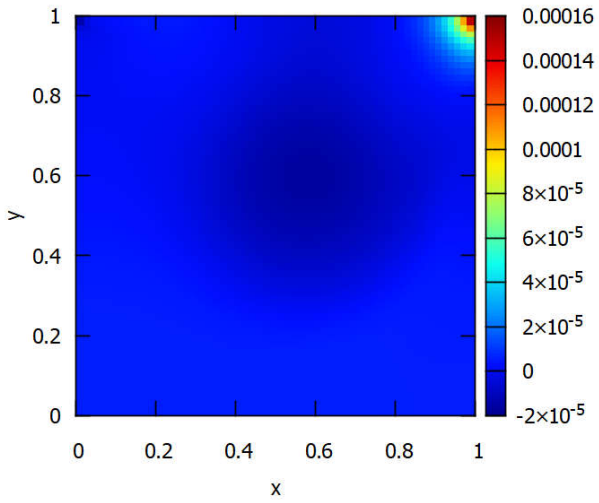



*Figure 37: Pressure distribution Re=1000*

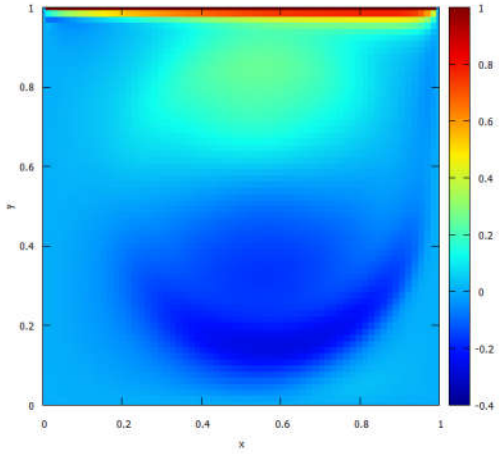



*Figure 38: x Velocity distribution Re=3200*

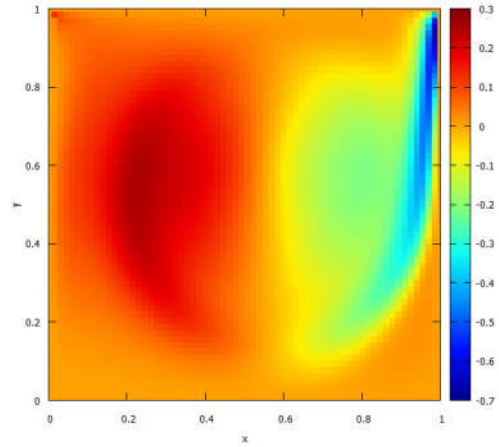



*Figure 39: y Velocity distribution Re=3200*

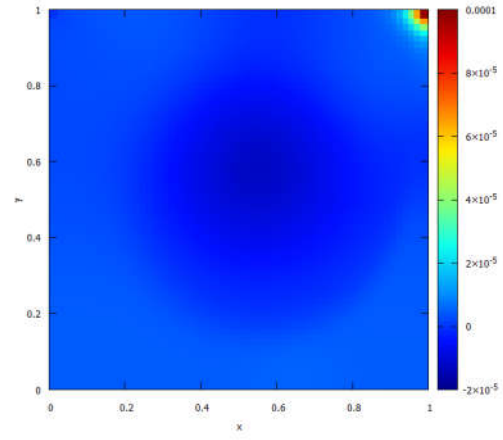



*Figure 40: Pressure distribution Re=3200*

Also, the velocity profile at the horizontal centreline is compared to the benchmark values

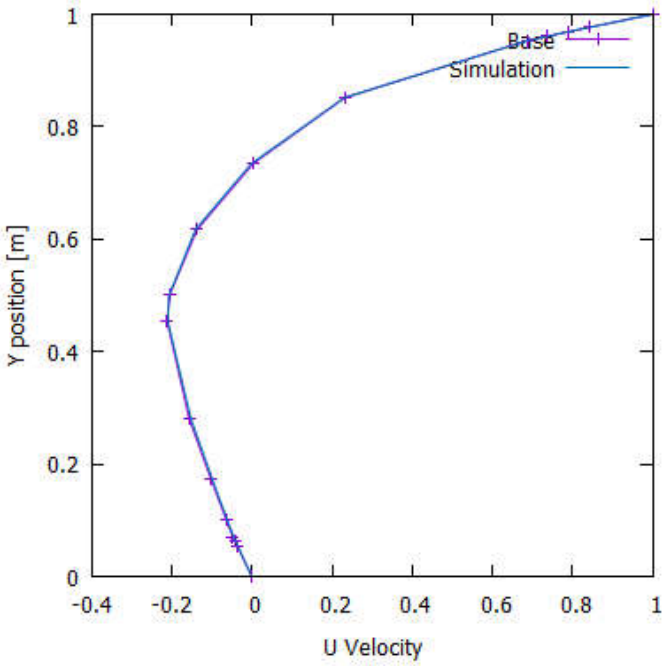



*Figure 41: x Velocity at vertical centreline Re=100*

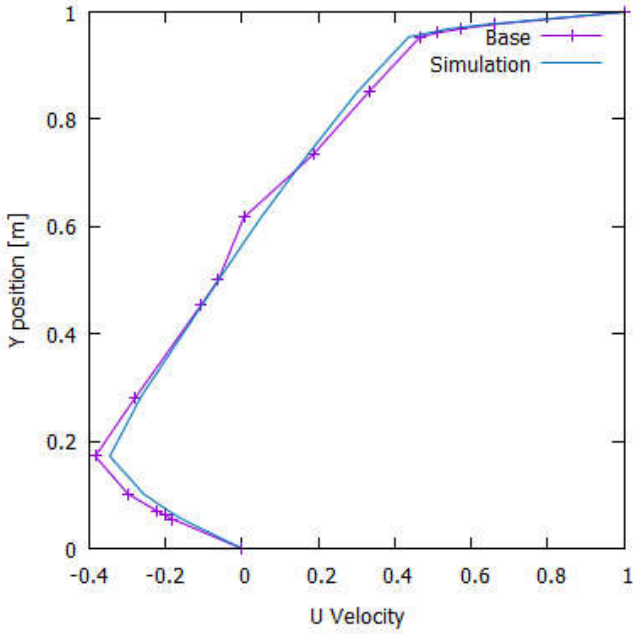



*Figure 42: x Velocity at horizontal centreline Re=1000*

### 3.4 DIFFERENTIALLY HEATED CAVITY

The simulation of a natural convection of an incompressible flow within a square cavity as shown in the figure below is presented. This is a variation of the Driven cavity problem (forced convection) which has been solved in the previous section. In the differentially heated cavity problem, movement of the fluid arises due to temperature differences in the cavity.

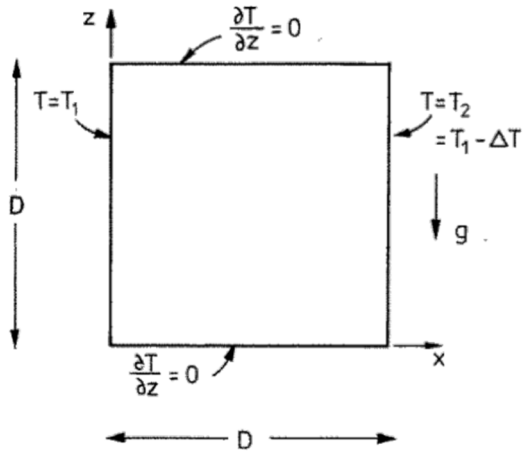



*Figure 43: Differentially Heated Cavity*

Furthermore, the energy equation is now introduced to the Navier-Stokes equation and the momentum equation is modified such that we have that

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

$$\frac{\partial(\mathbf{v})}{\partial t} + (\mathbf{v} \cdot \nabla)\mathbf{v} = Pr\Delta\mathbf{v} - \frac{1}{\rho_o}\nabla p + \boldsymbol{f}$$

$$\frac{\partial(\mathrm{T})}{\partial t} + (\mathbf{v} \cdot \nabla)\mathrm{T} = \alpha\Delta\mathrm{T}$$

Following the dimensional analysis presented in chapter 2, the equations are written thus

$$\nabla \cdot \tilde{\mathbf{v}} = \mathbf{0}$$

$$\frac{\partial(\tilde{\mathbf{v}})}{\partial \tilde{t}} + (\tilde{\mathbf{v}} \cdot \tilde{\nabla})\tilde{\mathbf{v}} = Pr\tilde{\Delta}\tilde{\mathbf{v}} - \tilde{\nabla}\tilde{p} + RaPr\tilde{T}\boldsymbol{a}$$

$$\frac{\partial(\tilde{\mathbf{T}})}{\partial \tilde{t}} + \tilde{\mathbf{v}} \cdot \tilde{\nabla}\tilde{\mathrm{T}} = \tilde{\Delta}\mathrm{T}$$

For simplicity sake, the accents that identify the dimensionless variables shall be dropped. A C++ program based on FVM is created to solve the problem with a wide range of Rayleigh number.

### 3.41 Fractional Step Method

The FSM, as explained in the driven cavity problem (3.3), is employed in this problem. However, rather than explain again, the necessary parameters peculiar to this case shall be included.

$\mathbf{R}(\mathbf{v})$is modified such that it becomes

$$\boldsymbol{R}(\mathbf{v}) \equiv -(\mathbf{v}\cdot\nabla)\mathbf{v} + Pr\Delta\mathbf{v} + RaPrT\boldsymbol{a}$$

Thus, the FSM for the differentially heated cavity is summarized as follows;

- Evaluation of $\boldsymbol{R}(\mathbf{v}) \equiv -(\mathbf{v}\cdot\nabla)\mathbf{v} + Pr\Delta\mathbf{v} + RaPrT\boldsymbol{a}$
- Evaluation of predictor velocity predictor $\mathbf{v}^p = \mathbf{v}^n + \Delta t\left(\frac{3}{2}\mathbf{R}(\mathbf{v}^n) - \frac{1}{2}\mathbf{R}(\mathbf{v}^{n-1})\right)$
- Evaluation of $\nabla\cdot\mathbf{v}^p$ and solve the Poisson equation $\Delta\tilde{p} = \nabla\cdot\mathbf{v}^p$ , this gives the distribution of the Pseudo pressure $\tilde{p}$
- Velocity of the next instant time $\mathbf{v}^{n+1} = \mathbf{v}^p - \Delta\tilde{p}$

### 3.42 Staggered Meshes

Also, the staggered mesh is the same as explained in section 3.3. Scalar fields such as pressure and temperature are evaluated in the main mesh while the component vectors in x and y are computed on the x-staggered and y-staggered meshes respectively.

### 3.43 Discretization

**Evaluation of** $R(\mathbf{v})$

Recall that $\boldsymbol{R}(\mathbf{v}) = -(\mathbf{v}\cdot\nabla)\mathbf{v} + Pr\Delta\mathbf{v} + RaPrT\boldsymbol{a}$

Resolving the velocity into x and y components;

$$\boldsymbol{R}(u) = -(\mathbf{v}\cdot\nabla)u + Pr\Delta u + RaPrTdxdy$$

But because g is zero in the x-direction the las term becomes zero and it becomes

$$\boldsymbol{R}(u) = -(\mathbf{v}\cdot\nabla)u + Pr\Delta u$$

$$\boldsymbol{R}(v) = -(\mathbf{v}\cdot\nabla)v + Pr\Delta v + RaPrTdxdy$$

Therefore, applying the gauss theorem and integrating $R(v)$ over a CV in the y-mesh, we have that

$$\int_{\mathrm{v}} \boldsymbol{R}(v)\mathrm{dV} = \int_{\mathrm{V}} (-(u\cdot\nabla)v + Pr\Delta v)\mathrm{dV} + RaPrTdxdy$$

$$-\int_{\mathrm{V}} \nabla\cdot(\mathbf{v}v)\mathrm{dV} + \mathrm{Pr}\int_{\mathrm{V}} \nabla\cdot(\nabla u)dV + RaPrTdxdy$$

$$-\oint_{dv} v\mathbf{v}\cdot\hat{n}dS + Pr\oint_{dv} \nabla v\cdot\hat{n}dS + + RaPrTdxdy$$

Approximating the surface integral across the four faces east, west, north and south, we obtain

$$\int_{\mathrm{V}} \boldsymbol{R}(v)\,\mathrm{d}V = \left(-v_e(u_e, v_e)\begin{pmatrix} n_{xe} \\ n_{ye} \end{pmatrix} S_e + Pr\left(\left.\frac{\partial v}{\partial x}\right|_e, \left.\frac{\partial v}{\partial y}\right|_e\right)\begin{pmatrix} n_{xe} \\ n_{ye} \end{pmatrix} S_e\right)$$
$$+ \left(-v_w(u_w, v_w)\begin{pmatrix} n_{xw} \\ n_{yw} \end{pmatrix} S_w + Pr\left(\left.\frac{\partial v}{\partial x}\right|_w, \left.\frac{\partial v}{\partial y}\right|_w\right)\begin{pmatrix} n_{xw} \\ n_{yw} \end{pmatrix} S_w\right)$$
$$+ \left(-v_n(u_n, v_n)\begin{pmatrix} n_{xn} \\ n_{yn} \end{pmatrix} S_n + Pr\left(\left.\frac{\partial v}{\partial x}\right|_n, \left.\frac{\partial v}{\partial y}\right|_n\right)\begin{pmatrix} n_{xn} \\ n_{yn} \end{pmatrix} S_n\right)$$
$$+ \left(-v_s(u_s, v_s)\begin{pmatrix} n_{xs} \\ n_{ys} \end{pmatrix} S_s + Pr\left(\left.\frac{\partial v}{\partial x}\right|_s, \left.\frac{\partial v}{\partial y}\right|_s\right)\begin{pmatrix} n_{xs} \\ n_{ys} \end{pmatrix} S_s\right) + RaPrTdxdy$$

Noting that

$S_e = S_w = \Delta y, S_n = S_s = \Delta x, \begin{pmatrix} n_{xe} \\ n_{ye} \end{pmatrix} = (1,0), \begin{pmatrix} n_{xw} \\ n_{yw} \end{pmatrix} = (-1,0), \begin{pmatrix} n_{xn} \\ n_{yn} \end{pmatrix} = (0,1),$

$\begin{pmatrix} n_{xs} \\ n_{ys} \end{pmatrix} = (0,-1)$

$$\boldsymbol{R}(v)dxdy = -v_e^* u_e \Delta y + Pr\frac{v_E - v_P}{\delta x}\Delta y + v_w^* u_w \Delta y + Pr\frac{v_W - v_P}{\delta x}\Delta y$$
$$- v_n^* v_n \Delta x + Pr\frac{v_N - v_P}{\delta y}\Delta x + v_s^* v_s \Delta x + Pr\frac{v_S - v_P}{\delta y}\Delta x + RaPrTdxdy$$

and

$$\boldsymbol{R}(u)dxdy = -u_e^* u_e \Delta y + Pr\frac{u_E - u_P}{\delta x}\Delta y + u_w^* u_w \Delta y + Pr\frac{u_W - u_P}{\delta x}\Delta y$$
$$- u_n^* v_n \Delta x + Pr\frac{u_N - u_P}{\delta y}\Delta x + u_s^* v_s \Delta x + Pr\frac{u_S - u_P}{\delta y}\Delta x$$

Noting that $u_f \; and \; v_f$ are the velocities of the corresponding face of the control volume obtained by simple interpolation while $u_f^* \; and \; v_f^*$ are the same velocities obtained by convective schemes. $u_F \; and \; u_F$ are the velocities at the neighbour points of the control point in the perpendicular to the faces.

**Evaluation of $\mathbf{v}^p$**

Given by the equation;

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

It can be expressed in the x-component and y-component as follows;

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

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

**Pressure Equation**

The pressure equation is similar to the one discretised in the driven cavity problem;

$$\frac{\tilde{p}_E - \tilde{p}_P}{\delta_{x_E}}\Delta y + \frac{\tilde{p}_W - \tilde{p}_P}{\delta_{x_W}}\Delta y + \frac{\tilde{p}_N - \tilde{p}_P}{\delta_{y_N}}\Delta x + \frac{\tilde{p}_S - \tilde{p}_P}{\delta_{y_S}}\Delta x = u_E^p\Delta y - u_W^p\Delta y + v_N^p\Delta x - v_S^p\Delta x$$

The discretized pseudo-pressure equation is thus given as

$$a_P\tilde{p}_P = a_N\tilde{p}_N + a_S\tilde{p}_S + a_E\tilde{p}_E + a_W\tilde{p}_W + b$$

The coefficients $a_P, a_N, a_S, a_E, a_W$ and $b$ are defined as follows;

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

$$a_N = \frac{\Delta x}{\delta y};$$

$$a_S = \frac{\Delta x}{\delta y}$$

$$a_E = \frac{\Delta y}{\delta x};$$

$$a_W = \frac{\Delta y}{\delta x};$$

$$b = \Delta y\left(-u_E^p + u_W^p\right) + \Delta x\left(-v_N^p + v_S^p\right)$$

**Velocity at next time-Step**

Given by the equation;

$$\mathbf{v}^{n+1} = \mathbf{v}^p - \nabla\tilde{p}$$

It is expressed in the x-component and y-component as follows;

$$u^{n+1} = u^p - \frac{\partial\tilde{p}}{\partial x}$$

$$v^{n+1} = v^p - \frac{\partial\tilde{p}}{\partial y}$$

Note that $\tilde{p} = p\Delta t$

**Temperature Equation**

For this case, temperature is included and the $\mathbf{R}(\mathrm{T}^n)$ is defined thus

$$\boldsymbol{R}(T) = -T_e^*u_e\Delta y + \frac{T_E - T_P}{\delta x}\Delta y + T_w^*u_w\Delta y + \frac{T_W - T_P}{\delta x}\Delta y - T_n^*v_n\Delta x + \frac{T_N - T_P}{\delta y}\Delta x + T_s^*v_s\Delta x + \frac{T_S - T_P}{\delta y}\Delta x$$

Noting that $u_f$ $and$ $v_f$ are the velocities of the corresponding face of the control volume obtained by simple interpolation while $T_f^*$ is the same Temperature obtained by CDS scheme. $T_F$ is the temperature at the neighbour points of the control point in the perpendicular to the faces.

Finally, the temperature at the next time step is evaluated as follows;

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

### 3.44 Boundary Conditions

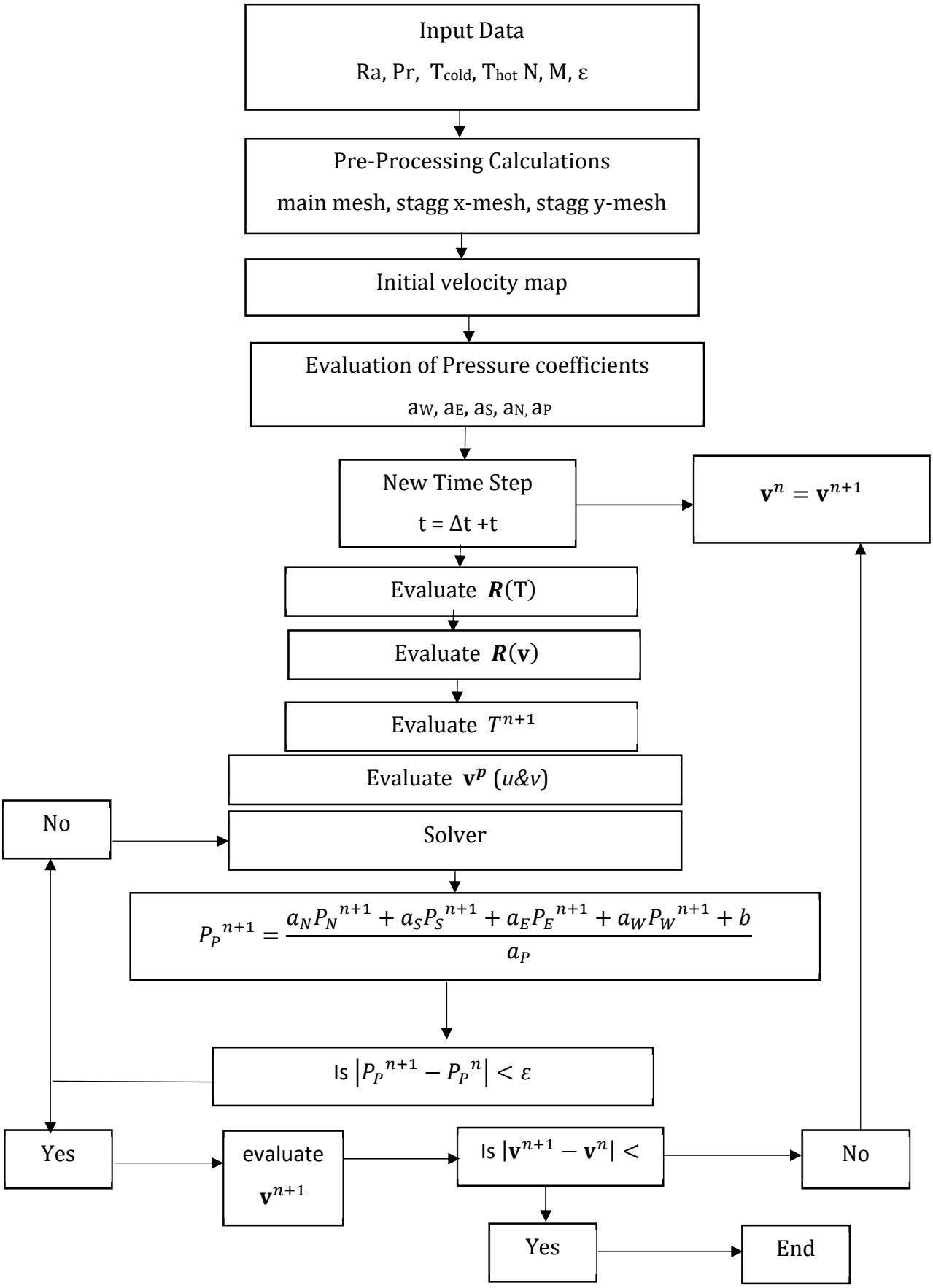
The boundary conditions of the heated cavity are summarized thus;

| **Boundary** | $u$ | $v$ | T |
|---|---|---|---|
| Top | 0 | 0 | adiabatic |
| Bottom | 0 | 0 | adiabatic |
| Right | 0 | 0 | 1 |
| Left | 0 | 0 | 0 |

*Table 7: Boundary condition (Differentially heated cavity)*

For the pressure equation, since it was given that $\partial p/\partial n$=0, we have that the pressure at the boundary is equal to the pressure of the neighbour node in the direction perpendicular to the boundary. For example, for any point [i][j], lying on the right boundary, we have that p[i][j] = p[i-1] [j]. And it is extended to other boundaries.

### 3.45 Algorithm



### 3.46 Results

Taking $Pr = 0.71\ (air)$ and $Ra = \{10^3, 10^4, 10^5, 10^6\}$, the differentially heated cavity problem was simulated. A C++ program was created and the to run the calculation. Using a grid size of $nx = ny = 50$ , distributions of x-Velocity, y-Velocity and Temperature (all non-dimensional) for the given range of $Ra$ are presented a follow;

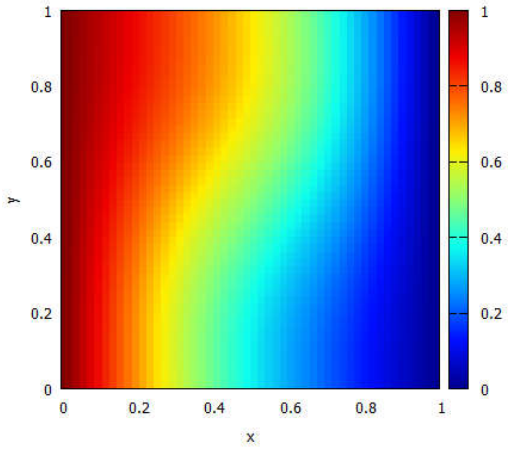



*Figure 44: Temperature, Ra=1e3*

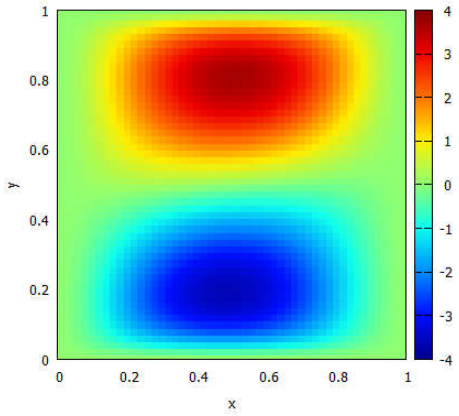



*Figure 45: Horizontal Velocity, Ra=1e3*

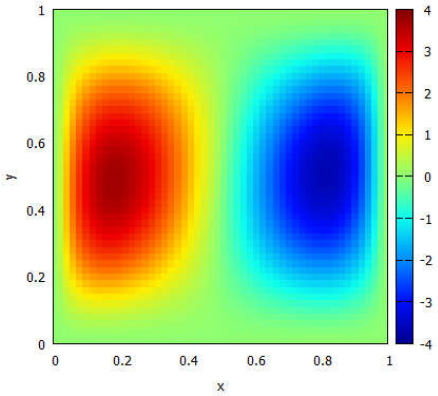



*Figure 46: Vertical velocity, Ra=1e3*

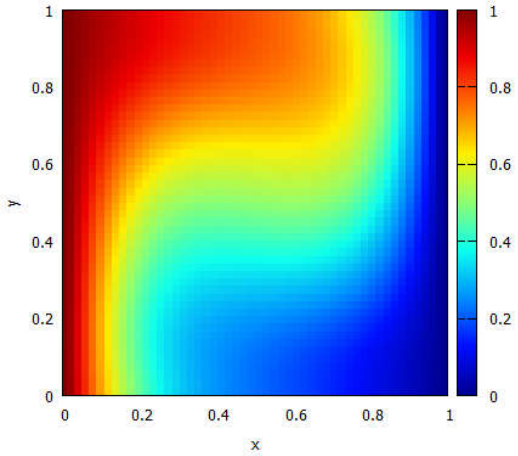



*Figure 47: Temperature, Ra=1e4*

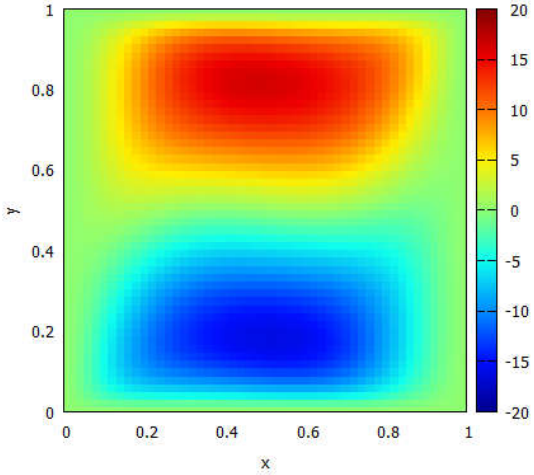



*Figure 48: Horizontal Velocity, Ra=1e4*

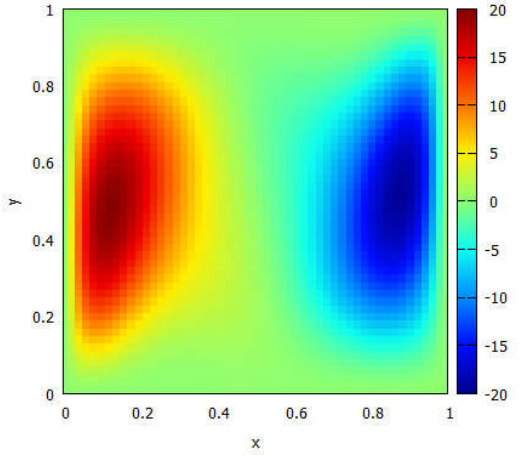



*Figure 49: Vertical velocity, Ra=1e4*

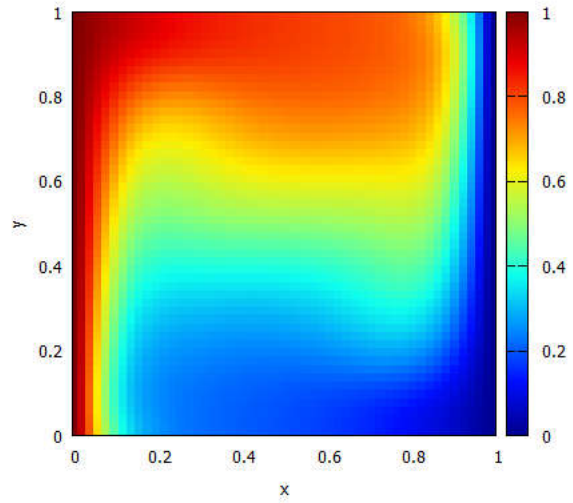



*Figure 50: Temperature, Ra=1e5*

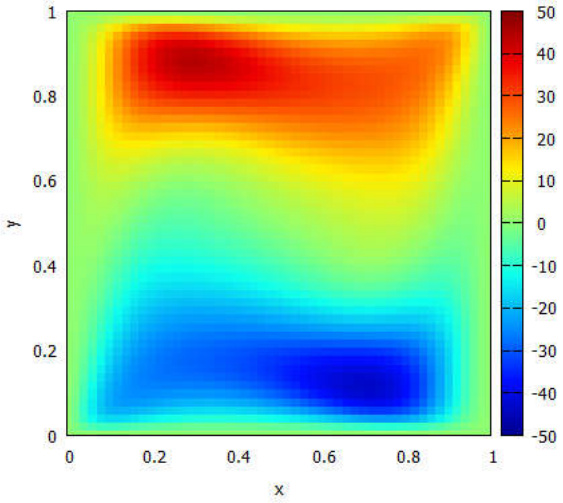



*Figure 51: Horizontal Velocity, Ra=1e5*

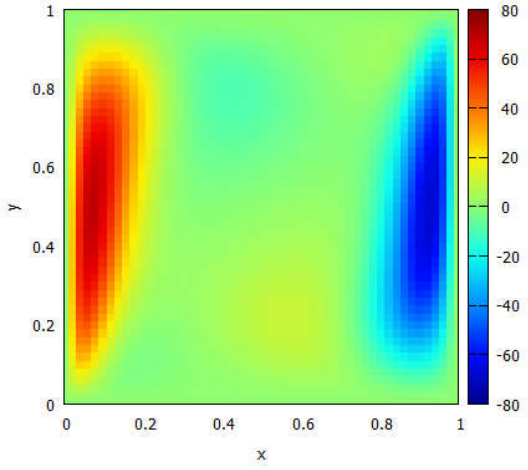



*Figure 52: Vertical velocity, Ra=1e5*

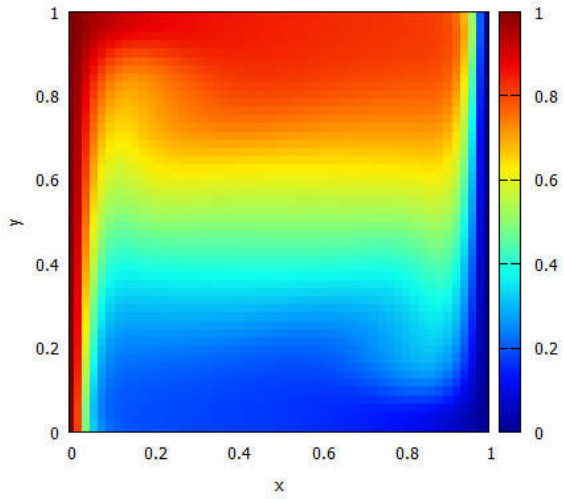



*Figure 53: Temperature, Ra=1e6*

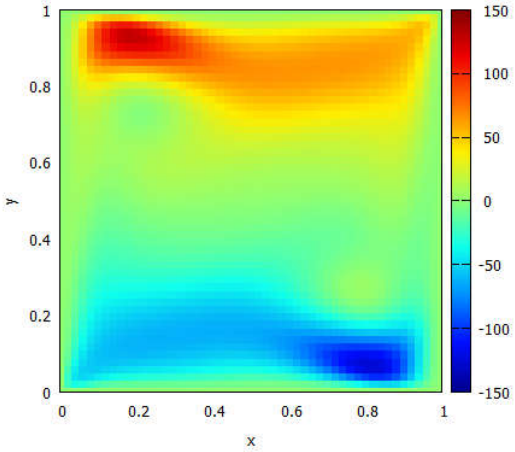



*Figure 54: Horizontal Velocity, Ra=1e6*

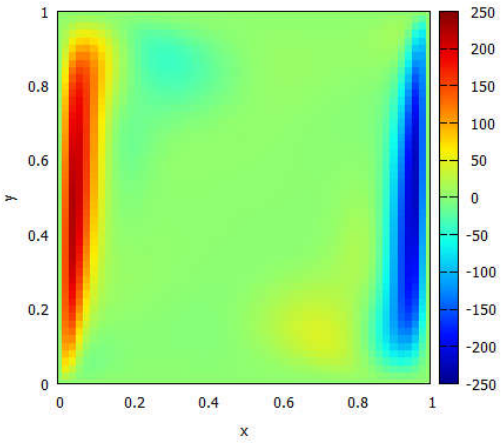



*Figure 55: Vertical velocity, Ra=1e6*

## 3.5 BURGER'S EQUATION

Despite its simplicity, important aspects of the 3D Navier-Stokes equation remain[12]. It is important to understand the process of energy transport in a turbulent flow. The Burger's equation offers this possibility. It can be considered as a model to understand some of the inside behaviour of the general problem.

The 1-dimensional case of the Burger's is resolved using the spectral numerical method which is done in the Fourier space. Simulation for both DNS and LES are presented for several modes and *Re.*

### 3.51 Discretization

Consider the non-dimensional form of Navier-Stokes equation;

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

We arrive at the burgers equation by substituting the last term with a forcing term **f**, and if follows that

$$\frac{\partial(\mathbf{v})}{\partial t} + \ (\mathbf{v}\cdot\nabla)\mathbf{v} = \frac{1}{Re}\Delta\mathbf{v} + \mathrm{f}$$

In a 1D space the equation becomes

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

Using the spectral method and applying the discrete Fourier decomposition, the velocity function is written thus;

$$u(x,t) = \sum_{-\infty\le k\le\infty} \hat{u}_k(t)e^{ikx} \cong \sum_{k=-N}^{k=N} \hat{u}_k(t)e^{ikx}\,, where\ \hat{u}_k(t)\ \epsilon\ \mathbb{C}$$

Finally, in Fourier space the burger's equation is written as;

$$\frac{\partial\hat{u}_k}{\partial t} + \sum_{p+q=k} \hat{u}_p i\hat{u}_q = \frac{1}{Re}k^2\hat{u}_k + \mathrm{F}_k, \qquad where\ k\ \epsilon\ \{-N,\dots,N\}$$

It can be deduced from the equation that;

1. There is better solution with increasing number of N
2. The convective term is responsible of the triadic interactions, since for a determined scale *k* the scales p and q appear.
3. The source term is responsible of maintaining the motion, otherwise the equation can be satisfied for all $\hat{u}_k = 0$.

4. The velocity $u(x,t) \in \mathbb{R}$ and therefore the condition $\hat{u}_k = \overline{\hat{u}_{-k}}$ must be accomplished ($\bar{\cdot}$ denotes the complex conjugate).

### 3.52 Boundary Condition

In the Fourier space, the forcing term $F_k$ will assume the role of the boundary condition which shall be imposed on the equation. It is presented as follows

$$F_k = \sum_{p+q=k} \hat{u}_p i\hat{u}_q + \frac{1}{Re} k^2 \hat{u}_k \;\; if \; k = 1$$

$$F_k = 0 \;\; if \; k \neq 1$$

Also, if we integrate the discretized Burger's equation over time step Δt and apply an explicit scheme, we arrive at the following expressions;

$$\hat{u}_k^{n+1} = \hat{u}_k - \Delta t(\hat{Conv}_k + \hat{Diff}_k) \;\; \text{if k} \neq 1$$

$$\hat{u}_k^{n+1} = \hat{u}_k \;\; \text{if k} = 1$$

Where

$$\hat{Conv}_k = \sum_{p+q=k} \hat{u}_p i\hat{u}_q$$

$$\hat{Diff}_k = \frac{1}{Re} k^2 \hat{u}_k$$

**LES**

The above equation is used for the DNS approach. For the LES model, the diffusive term is modified using the spectral eddy-viscosity model proposed by Krachian and improved by Metais and Lesieur[12] . It is modified as follows;

$$\hat{Diff}_k = \nu k^2 \hat{u}_k$$

$$\nu = \nu_t(k) + Re^{-1}$$

$$\nu_t(k) = \nu_t^{+\infty} \left(\frac{E_{k_N}}{k_N}\right)^{0.5} \nu_t^* \left(\frac{k}{k_N}\right)$$

With

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

Where $m$ is the slope of the energy spectrum, that is $k^{-m}$, $E_{k_N}$ is the energy at the cut-off frequency at $k_N$, $C_k$ is the Kolmogorov constant. $\nu_t^*$ is a non-dimensional eddy-viscosity

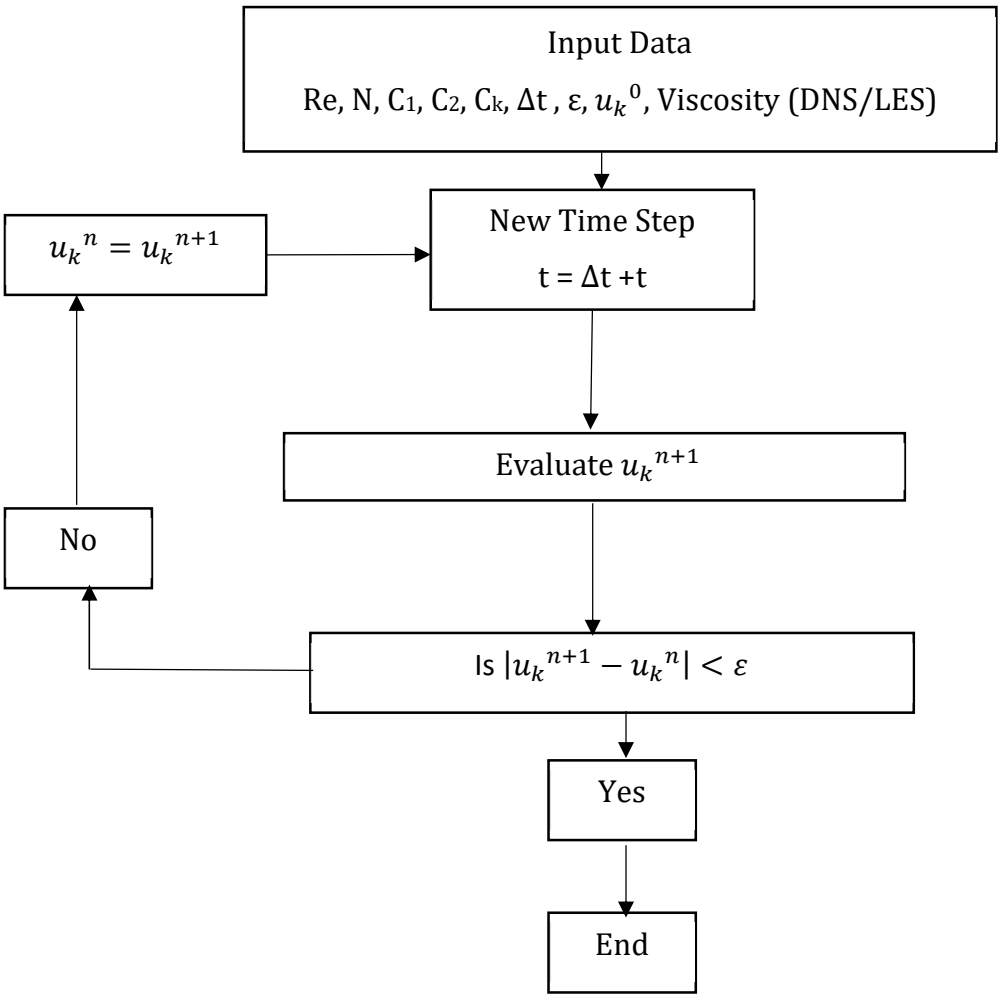
which is 1 for small values of $\frac{k}{k_N}$ and with a strong increase for higher $k$ up to $\frac{k}{k_N} = 1$. It is expressed thus

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

A CFL-like condition was be imposed to determine the size of the time step (since a fully explicit time-integration scheme is being used).

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

### 3.53 Algorithm



### 3.54 Results

With an initial condition of $\hat{u}_k = k^{-1}\ k = 1, \ldots, N$ and $\hat{u}_0 = 0$; $C_k = \{0.4223, 0.05\}$; $Re = \{10,40,100,400\}$; $N = \{40, 100\}$, a C++ program is created to implement the resolution of 1D burgers equation using the spectral method. The plot of the energy spectrum $E_k$ against frequency k for Re=40 and $C_1$=0.0001 is presented below;

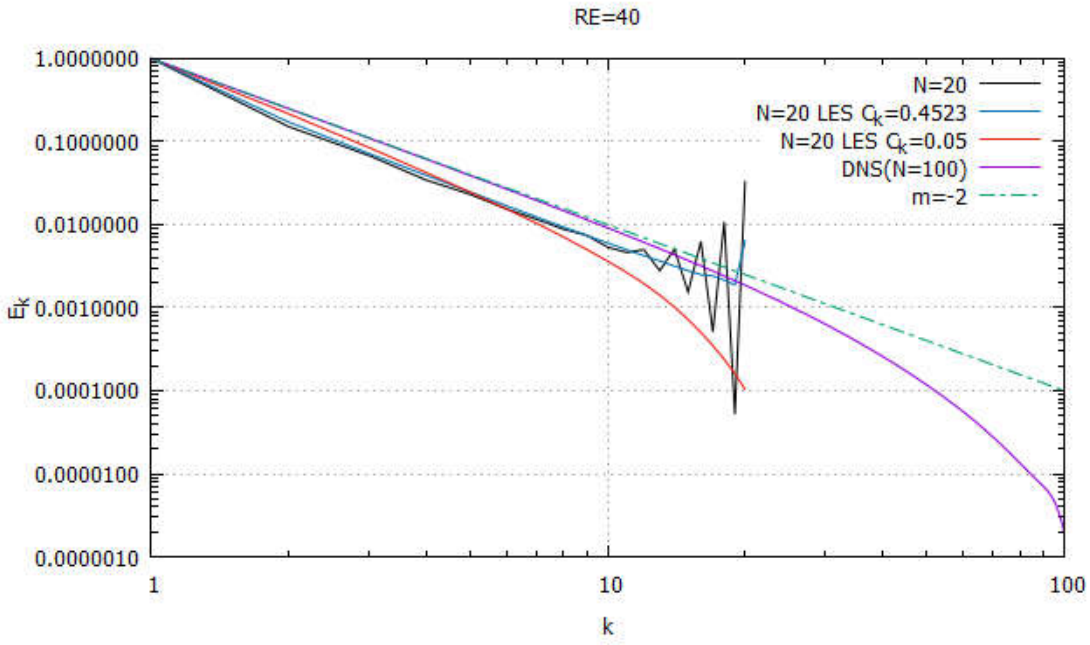



*Figure 56: Re=40*

From Figure 56, it is observed that the DNS for N=100 (fine mesh) gives a precise solution while simulation for N=20 (coarse mesh) does not converge. This is consequent on the inability of coarse mesh to simulate energy dissipated in small scales.

Also, the convective term of the equation is responsible for the transfer of kinetic energy from large scales (low frequency modes) to small scales (high frequency modes). This can also happen vice-verse. A phenomenon caused by Energy backscattering.

It is also interesting to note that there exists an energy damping effect on the convective term by the diffusive term. This is more prevalent in modes of higher frequencies

**Variation in Reynolds Number**

The energy spectrum of the steady-state solution of the Burger's equation with a time-step factor of $C_1$=0.0001 with varying values of Reynolds number are presented below;

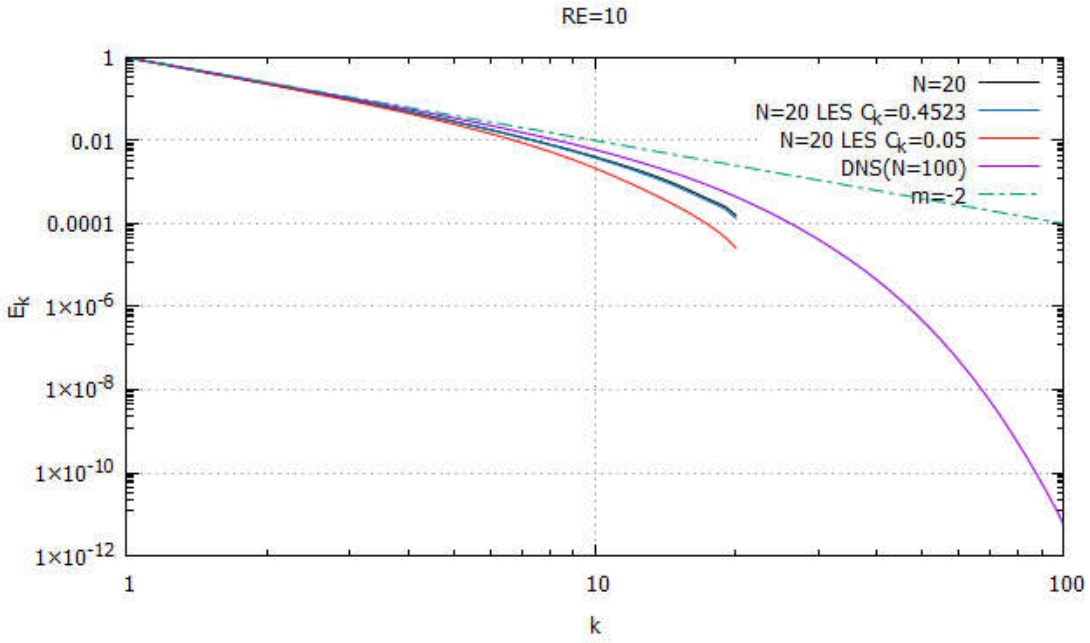



*Figure 57: Re=10*

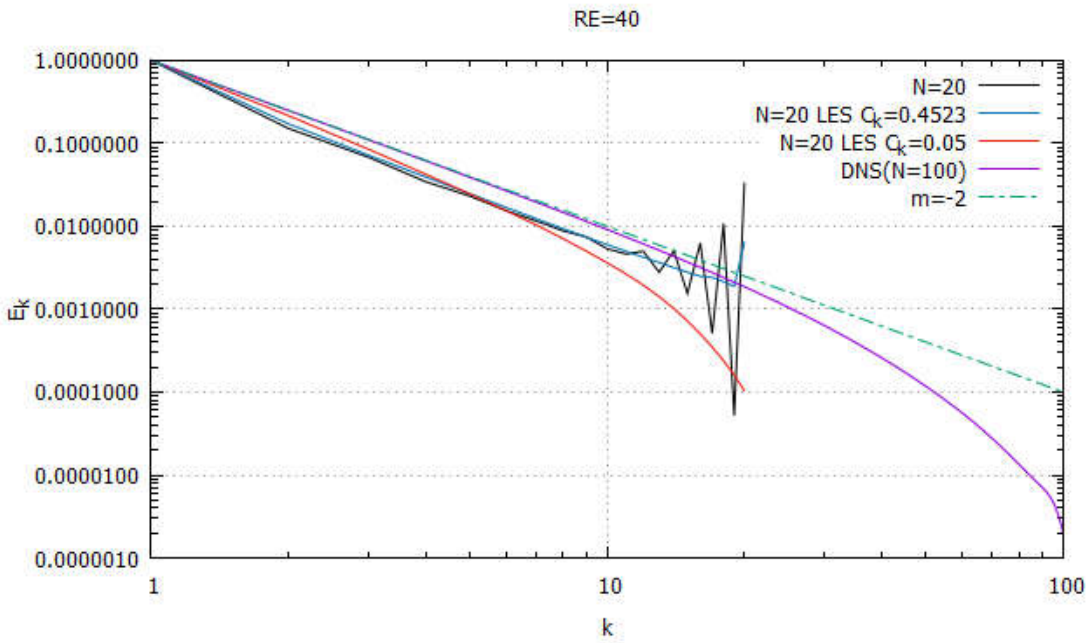



*Figure 58: Re=40*

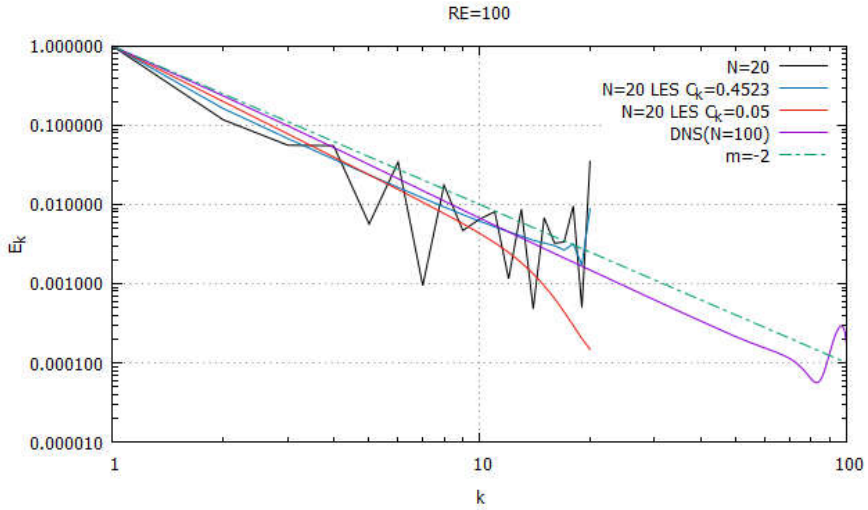



*Figure 59: Re=100*

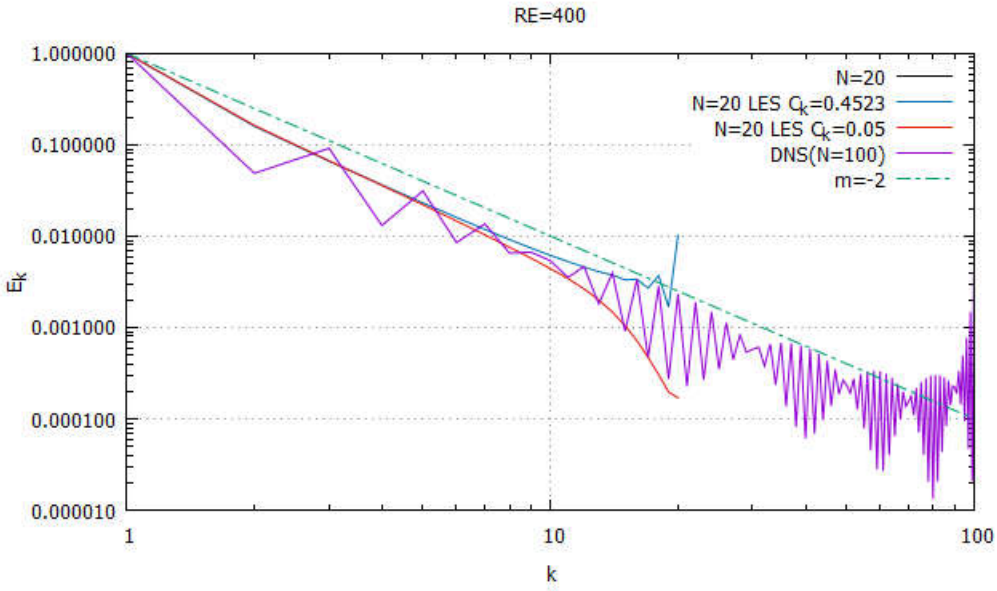



*Figure 60: Re=400*

From the results presented above the following can be deduced;

1. Increase in Re leads to a decrease in the influence of the diffusive term, while the convective term increases.
2. Higher Re requires finer mesh for accuracy and precision. This explains the divergence of DNS simulation of N=20 (Figure 60). However, this also a good representation of effect introducing other turbulence models like the LES, which helps to compensate this phenomenon.

3. The Large-Eddy Simulation is a fascinating way to optimize the process giving good results also for low number of modes. This is achieved by the ability of LES to apply extra numerical diffusion.

**Effect of Time-Step**

For Re=40, simulations were carried out for different time-step. The time-step is characterized by changes in values of $C_1$. Results for values $C_1$ =0.1, 0.01, 0.001 and 0.0001 are presented as follows;

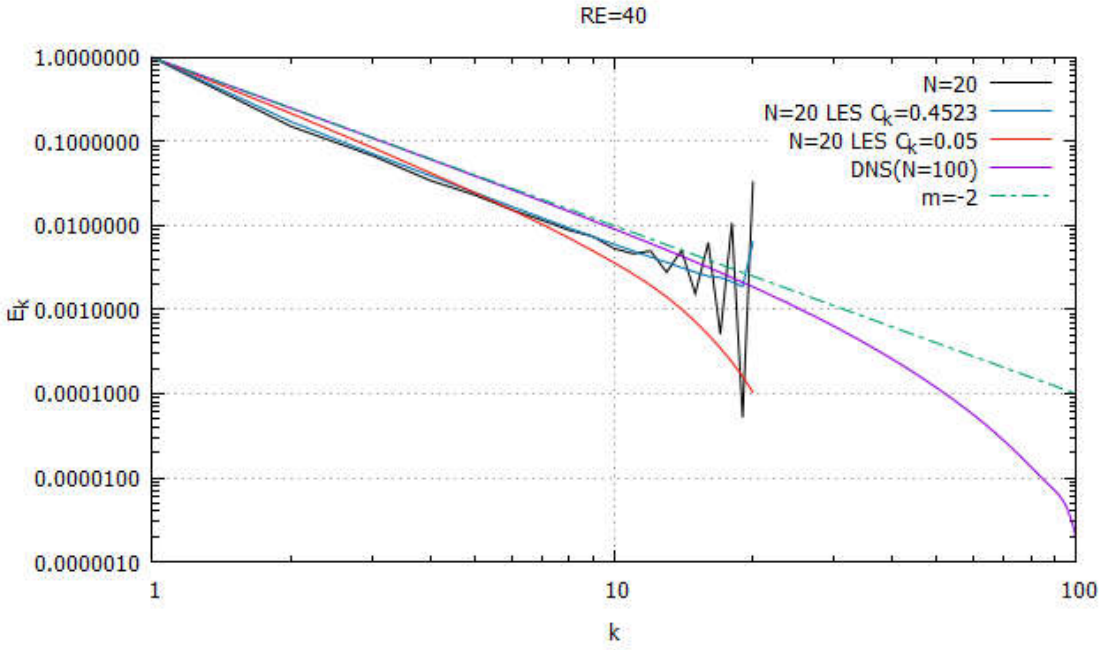



*Figure 61: C1= 0.0001*

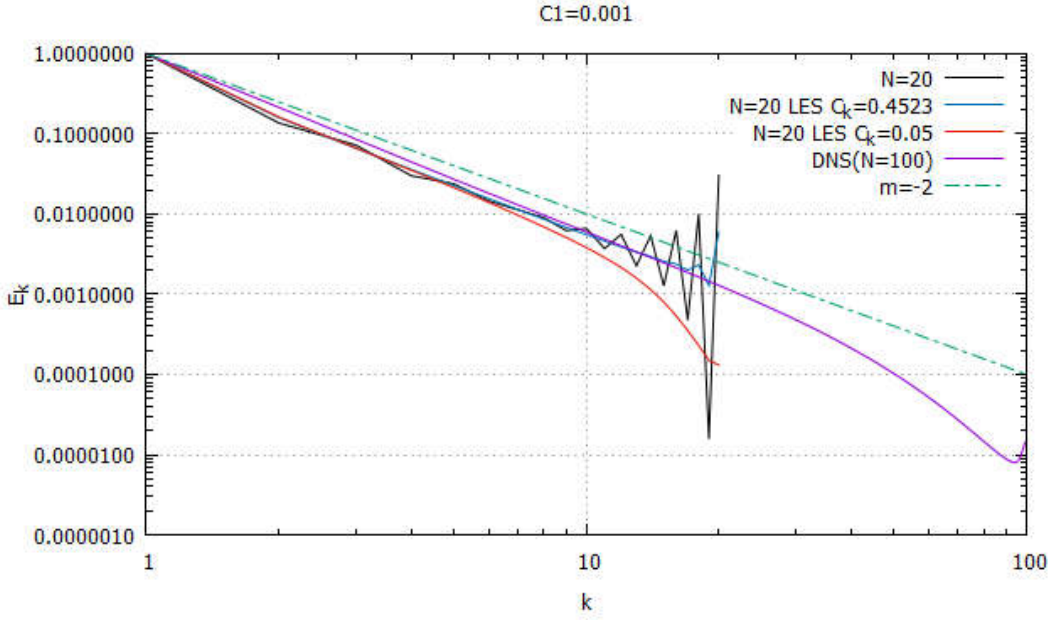



*Figure 62: C1= 0.001*

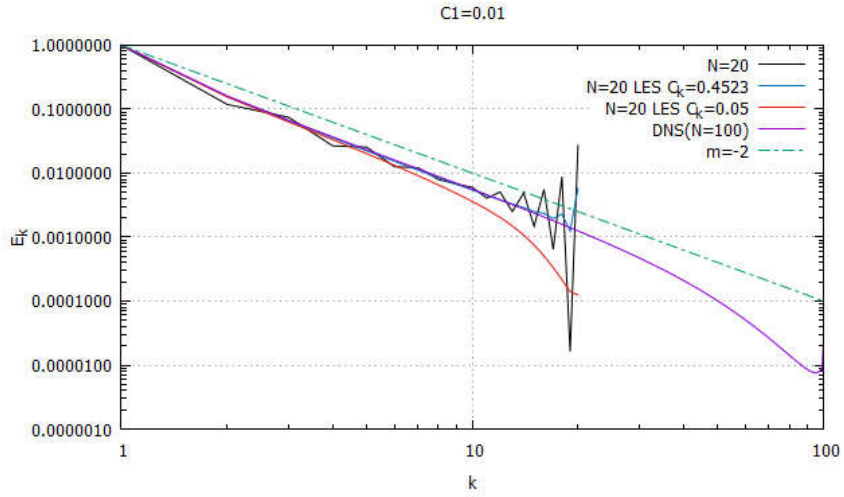



*Figure 63: C1=0.01*

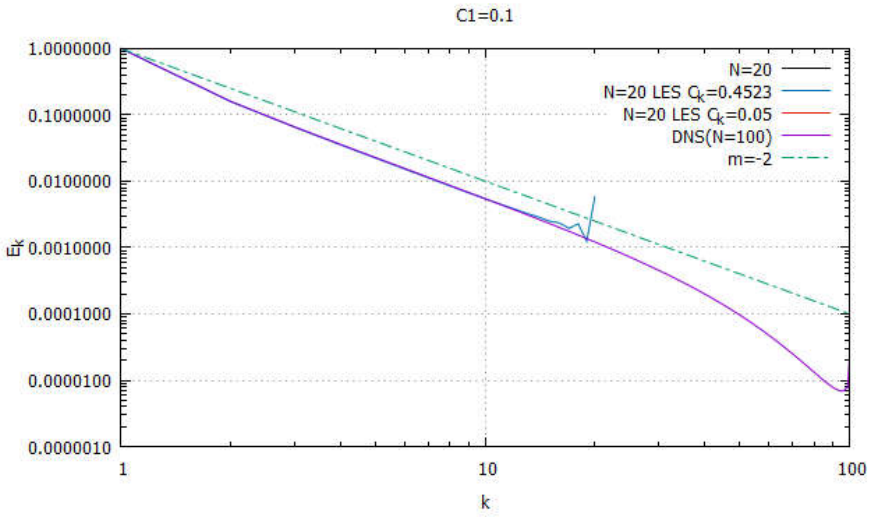



*Figure 64: C1=0.1*

From the results above, it can be realized that:

1. Variation in time-step has little or no effect on DNS simulation for fine mesh.
2. Despite little improvements for coarse mesh at the least time-step, there is a little variation in results for changing time-step. However, at $C_1$=0.1, solution is not reached.
3. Precision decreases with increasing time-step while computational cost decreases with increasing time-step.

## CONCLUSION

The resolution of five distinct cases of fluid dynamics and heat tranter has been presented. The simulations reveal the importance of numerical methods and modelling for broad human activity. This, surely, will help save time and resources that cold have been lost in experiments and analysis in which accuracy are not guaranteed.

For the pure conduction case, the use of harmonic mean to find the conductivity at the interface of a control volume helps to dispel problems that may arise due to changing conductivity. And this may be extended to other cases of varying material properties.

The convection-diffusion cases explored shows that high-order schemes offer better accuracy than low order schemes.

Fractional step method has been shown to play a vital role in solving the checker-board problem. Also, increase in Reynolds number is seen to increase the computational costs.

As in the solution of the Burger's equation, selecting the proper time-step is a prerequisite to obtaining a good result. Noting that introducing other turbulence models like LES helps to overcome challenges that are present in the use of DNS.

Finally, it would be interesting to implement the LES to the Driven cavity problem and compare the results with DNS.

## REFRENCES


[1] S. C. Chapra and R. P. Canale, “Numerical Methods for Engineers.” McGraw-Hill Science/Engineering/Math, 2015.

[2] S. V. Patankar, “Numerical Heat Transferand Fluid Flow.” Hemishphere Publishers, New York, p. 197, 1980.

[3] F. Moukalled, L. Mangani, and M. Darwish, “The Finite Volume Method in Computational Fluid Dynamics,” vol. 113. 2016.

[4] D. M. Causon and C. . Mingham, “Introductory Finite Difference Methods for PDEs.” 2010.

[5] “Finite Element vs Finite Volume | CFD | Autodesk Knowledge Network.” [Online]. Available: https://knowledge.autodesk.com/support/cfd/learn-explore/caas/CloudHelp/cloudhelp/2014/ENU/SimCFD/files/GUID-12A9AED8-2047-4D3A-BC80-82BE9CF47517-htm.html. [Accessed: 27-Dec-2019].

[6] “What are Boundary Conditions? — SimScale Documentation.” .

[7] R. E. Kalaba and K. Spingarn, “A criterion for the convergence of the Gauss-Seidel method,” *Appl. Math. Comput.*, vol. 4, no. 4, pp. 359–367, Oct. 1978.

[8] H. Tennekes and J. Lumley, “A First Course in Turbulence,” *SERBIULA (sistema Libr. 2.0)*, 2014.

[9] A. N. Kolmogorov, “The Local Structure of Turbulence in Incompressible Viscous Fluid for Very Large Reynolds Numbers,” 1991.

[10] CTTC, “Large-Eddy Simulation of turbulent incompressible flows,” 2015.

[11] CTTC, “Introduction to the Fractional Step Method,” pp. 1–10.

[12] CTTC (UPC), “Burgers’ equation in Fourier space,” vol. 4, no. 4, pp. 1–8, 2014.