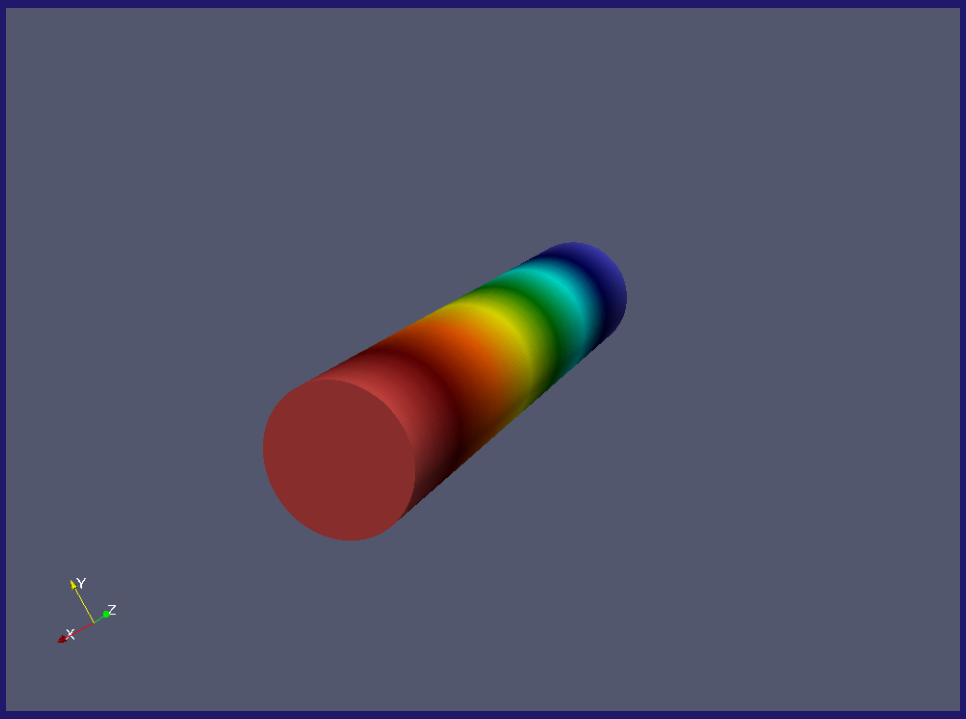
Modeling of turbulent two-phase stratified flow

Berkcan Kapusuzoğlu

Master of Science Thesis



TU Delft Delft University of Technology

Challenge the future

Modeling of turbulent two-phase stratified flow

by

Berkcan Kapusuzoğlu

in partial fulfillment of the requirements for the degree of

Master of Science

in Applied Mathematics

at the Delft University of Technology,

to be defended publicly on Monday November 28, 2016 at 10:00 AM.

Supervisor:	Dr. ir. D. R. van der Heul	
Thesis committee:	Prof. dr. ir. C. Vuik,	TU Delft
	Dr. J. L. A. Dubbeldam,	TU Delft

This thesis is confidential and cannot be made public until November 28, 2016.

TU Delft Delft University of Technology

Preface

The work described in this report has been written within the Erasmus Mundus Master's program Computer Simulations for Science and Engineering (COSSE [1]). This thesis project 'Modeling of turbulent two-phase stratified flow' is carried out at the Applied Mathematics Department of Delft University of Technology. It represents the culmination of all my efforts during my master study one year at the Friedrich-Alexander University Erlangen-Nuremberg, Germany, and one year at the Delft University of Technology, the Netherlands.

This study is performed under the supervision of Dr. ir. Duncan Roy van der Heul. I would like to thank him and also Ir. Guido Oud for giving me some useful feedback throughout the project.

Next, I would like to thank my family. Filiz, my mother, was always there for me with her unconditinal infinite love and support. Mehmet, my father, encouraged me throughout this phase of my life. The relationship that I formed and the memories I made with my grandparents, Nuran and Yılmaz, are invaluable to me. They have always supported me and motivated me with their pure love. Ali, my uncle, my aunt, Selda, and his husband, Hakan, always supported and galvanized me from a very far away land, Australia, with a sense of what I could yet do.

Special thanks to my dear Meltem. She entered my life during this master's program and mesmerized my life with her smile. I believe that we have become a better version of ourselves since the first day we met. Despite all the distance separating us, I have always felt her presence and support.

Finally, I would also like to thank all my friends with whom I have shared memories during the past two years and with some even more.

Berkcan Kapusuzoğlu
Delft, November 2016

[1] For more information: https://www.kth.se/en/studies/master/cosse

Abstract

The thesis is focused on the current state-of-the-art of modeling turbulent single-phase and turbulent two-phase stratified flows. This report is meant to identify and validate the most suitable candidate model for inclusion in the multiphase flow models that are being developed in the Scientific Computing group of the Delft Institute for Applied Mathematics. For that reason, as an initial step the results of single phase turbulent pipe flows which are simulated using DNS and LES methods are compared with the results of Eggels [1].

In two-phase flows, if the gas phase is turbulent the liquid phase will become turbulent as well. If the transition occurs from stratified flow to stratified-wavy flow, the interfacial momentum transfer varies due to the existence of waves at the interface. This process makes modeling of the momentum transfer complicated. In general, when the effects of surface tension are negligible the equations for the two-phase flow and for the single-phase flow are identical, the only differences between two-phase and single phase flow equations are the variable density and viscosity. Therefore, the influence of the interface and the momentum transfer between both phases can be ignored and a simple single-phase flow model combined with an interface model can be used as an initial approximation while concentrating on modeling the turbulence in both phases away from the interface. For this reason, the *one-fluid* model needs to be introduced (refer to Appendix E) in order to obtain results for multiphase flows using classical single-phase flow models. In this project, two different types of numerical techniques, namely DNS and LES are chosen to estimate the computational resources of a turbulent single-phase pipe flow test case with a friction Reynolds number of $Re_* = 360$. Accordingly, computational complexities of different techniques are analyzed in detail. The estimation procedure of the problem complexity (i.e., the required number of total grid cells) for turbulent single-phase flows gives an underestimate for the number of unknowns of turbulent two-phase flows. The comparison of computational costs showed that *Direct Numerical Simulation (DNS)* is possible for turbulent two-phase stratified pipe flows only for low Reynolds numbers. For high Reynolds number flows, DNS is not feasible because the current implementation of the algorithm is not parallelized and the computational resources of the Scientific Computing group are limited. Because of these reasons, *large eddy simulation (LES)* is considered to be the promising technique as the computational resources required for DNS become excessive for higher Reynolds number the serial code. Therefore, LES needs to be investigated elaborately for turbulent two-phase stratified pipe flows in future. At this point, several numerical simulations are performed to take first steps towards simulation of turbulent two-phase flows using the *one-fluid* model. However, it is due to limitations

in time, turbulent two-phase flow simulations are not performed and only turbulent single-phase flows are considered in this thesis.

The numerical results for the Poiseuille flow are obtained both in Cartesian and cylindrical coordinates to verify the variable viscosity formulation before analyzing turbulent flows. The algorithm developed in the Scientific Computing group is improved with necessary periodic boundary conditions for the discretization of the equations that describe turbulent single-phase flow in a circular pipe geometry. First, these boundary conditions are introduced into the algorithm and then results of numerical simulations in 2D (channel flow) and 3D (axisymmetric pipe flow) are validated by comparing them with theoretical values in Section 5. Then, the variable viscosity formulation is incorporated into the algorithm to take a step towards LES computations. In addition to this, the subgrid scale (SGS) parametrization and the Smagorinsky model are utilized for treatment of SGS turbulence which constitutes the basis of LES. The present numerical results for both DNS and LES illustrate that they are in agreement with the results of Eggels [1]. Moreover, both methods are capable of simulating the problem within a reasonable amount of time and accuracy. It is also shown that choosing a relatively smaller pipe length (because of the restrictions imposed by the serial code) than the one chosen by Eggels [1] has no significant effect on the resulting velocity profiles.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Background of multiphase flow

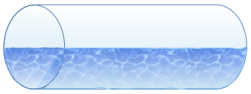
Any flow that consists of more than one fluid or a fluid and a solid is called a *multiphase flow*. Multiphase flow can be classified according to the state of phases that occur simultaneously in the flow domain such as gas and liquid flow, liquid and solid flow or gas and particle flow. If the state or the phase are the same, but the material properties are different (i.e. oil and water; liquid-liquid) for the flow, then the flow is also classified as a multiphase flow.

In general, multiphase flow has two general topologies: *disperse flow* and *separated flow*. Disperse flow consists of particles, drops or bubbles in the flow. However, in separated flow, as the name suggests, the streams of different fluids are separated by interfaces.

Almost every process technology has involvement with multiphase flow, thus, it occurs in many areas in industry, such as oil and gas recovery, (nuclear) power generation, food and chemical production. For safe transport and processing, the multiphase flow is required to be stable and predictable. Therefore, computational fluid dynamics (CFD) plays an important role at this point to simulate the environment and find the most cost-effective and efficient system design.

In long distance pipelines (e.g. steam and water or natural gas and oil flows), in power generation, petrochemical and process plants, the flow regime is so called stratified flow (Fig. 1.1a). With an increase in the flow rate of the top fluid, waves occur on the interface of the two fluids. The stratified flow in the pipeline first changes to stratified wavy flow (Fig. 1.1b), then to slug flow (Fig. 1.1c), as the gas velocity increases. If necessary precautions are not taken, these waves can get high enough to reach the top of the pipe. After that point, the gas flow can be blocked and the flow becomes discontinuous, which leads to formation of slugs (liquid slugs occur in the gas phase and the gas phase consists of large bubbles). These discontinuities may be engendered by discontinuous structures in the gas phase and by the abrupt turn in flow. This should be avoided at all times since it can lead to pressure fluctuations and damage in the pipeline system; especially at the

bends. Therefore, being able to predict the onset of the transition from wavy to slug flow is very important. In this study, modeling of immiscible incompressible *turbulent* two-phase



(a) Stratified flow

(b) Stratified wavy flow

(c) Slug flow

Figure 1.1: Gas-liquid flow regimes in horizontal pipes

stratified flow is investigated. The aim of this thesis is to make a start with turbulence modeling for stratified two-phase flow. Different models are compared in order to find the most appropriate model, which predicts the onset of instability of the interface and the formation of slugs. The main challenge for modeling turbulent two-phase flow is the turbulent conditions for bulk motion. Turbulence plays an important role in the transition of stratified flow to wavy flow. Moreover, waves on the interface have an influence on the dynamics of the interface, which leads to turbulence. Therefore, modeling turbulent two-phase flow is difficult compared to modeling turbulent single phase flow. There are quite many different models for turbulence in single-phase flow. However, not all of these models are extended to turbulent two-phase flow.

In the Laboratory for Aero- and Hydrodynamics of Faculty of Mechanical, Maritime and Materials Engineering of Delft University of Technology, experiments have been performed to predict the formation of slugs and the transition from laminar to turbulent flow. As a final goal, the configuration and flow parameters of the experiments can be utilized, and the obtained results can be validated with the results of the experiments. Moreover, in order to verify the methodology used in this thesis, the computational technique used in the work of Eggels [1] is analyzed in Appendices A and B. In that manner, the models are validated and limitations of all models are identified clearly.

In this thesis, first, single phase flow is simulated with DNS by introducing necessary boundary conditions. It is practical to use DNS for simulating two-phase flows within a reasonable amount of time with the available computational resources since the Reynolds number [1] of the flow is low enough ($Re_b = 5300$). For this reason, DNS is going to be used to compare the results with the work of Eggels [1]. However, the computational cost of DNS increases rapidly (with the Reynolds number to the 9/4th power) for high Reynolds number flows and the available resources and the serial code in the Scientific Computing group of the Delft Institute for Applied Mathematics are not sufficient enough to simulate a high Reynolds number single- or two-phase flows using DNS. Therefore, LES is realized as the most promising technique for turbulent two-phase flow.

[1] $Re_b = (\bar{u}_z)_b L/\nu$ is the Reynolds number for the bulk velocity, $(\bar{u}_z)_b$ is the mean velocity of the fluid, L is the characteristic length scale of the flow geometry (e.g. the pipe diameter), and ν is the kinematic viscosity of the fluid.

The modeling of multiphase turbulent flows compared to single-phase turbulent flows with high accuracy is more difficult. Although, the interface between two immiscible phases can be described quite precisely with available methods, the influence of the turbulent fluctuations in one of the phases may have great influence on the dynamics of the interface. Thus, it is very important to clarify the effect of the turbulence in all phases. There are considerably few studies about turbulence model of two-phase stratified pipe flow. In addition to this, LES is not the common practice in turbulent two-phase stratified pipe flow. Therefore, in order to have an idea about the flow properties (e.g. velocity profile, pressure drop, etc.), DNS and LES methods are used for single-phase turbulent pipe flows, then, necessary inferences are planned to be made about two-phase turbulent pipe flows. Due to limitations in time, two-phase turbulent pipe flow computations could not be carried out, thus, necessary inferences cannot be made about the two-phase turbulent pipe flows.

1.2 Problem description of multiphase flow modeling

The turbulent behavior near the interface becomes challenging to model accurately as the turbulence of the flow augmented with an increase in fluid velocities. Also, the turbulent flow near the interface affects the momentum transfer between the phases, which is the critical and peculiar phenomenon of turbulent two-phase flows.

Although, stratified flow is considered to be the simplest case of gas-liquid flow, the momentum transfer between the two phases is not completely understood. The difficult part is the formation of waves at the interface and the interaction between this deformed interface and the two fluids. Experimental studies have been carried out for stratified wavy gas-liquid flow. However, it is quite challenging to get an accurate result for the velocity profile close to the interface [3].

Instantaneous values for pressure and velocity of a fluid, which are governed by the Navier-Stokes equations, can be decomposed into a mean and a fluctuating part Eq. (2.5) using the *Reynolds decomposition* [2]. The continuity and the Navier-Stokes equations can be described only with the mean value by taking the average of the eqs. (2.7) and (2.8) in time. As a result of the averaging procedure, the *Reynolds Averaged Navier-Stokes (RANS)* equations are obtained and new unknown terms, *Reynolds stresses* appear in the equations, which need to be modeled (see Eq. (2.11)). This leads to a *closure problem*; the introduction of more equations and new unknown correlations. Moreover, when new equations are developed for these unknown terms, more unknown terms appear in the equations, which means that the number of unknowns is larger than the number of equations. In fact, the closure problem suggests that there is a need for infinite number of equations in order to describe the turbulence statistically.

The numerical methods for solving the governing equations and the closure problem for turbulent two-phase flows are quite complex. In most of the cases, two-phase flows show periodic behavior [4]. Furthermore, the waves on the interface have an effect of changing the flow from laminar to turbulent. More importantly, the turbulent fluctuations in each

[2] A mathematical technique that decomposes the instantaneous quantities into time-averaged and fluctuating quantities.

phases will influence the dynamics of the interface.

For single-phase flows, there are turbulence models available for most specific types of flows. However, it is not the case for turbulent two-phase flow since the momentum transfer at the interface cannot be handled easily [4]. The most common numerical approach for single-phase turbulent flows used in engineering applications is based on the RANS equations (see Eq. (2.9)), in which the effect of turbulence fluctuations are modeled. This approach yields different models, such as the two-equation models ($k-\epsilon$ model), which can be used to predict many flows that are fully turbulent except flows with strong separation, swirling, or rotation. Another model that rises from the application of Reynolds averaging is the Reynolds Stress Model (RSM), which can be used for free shear flows with strong anisotropy, flows with sudden changes in the mean strain rate.

Direct numerical simulation (DNS) is the most accurate and easy-to-implement numerical approach to the solution of turbulent flow. In DNS, all of the scales of turbulent motion are resolved in space and time explicitly. The range is from macro-structure scales (energy-carrying) to micro-structure scales (dissipative motions), which makes it a very costly method. The number of grid points is proportional to $Re_\tau^{9/4}$ (see Eq. (1.1)). Therefore, DNS is applicable only to simple geometries, and is limited to flows with low Reynolds numbers. It is often used to validate the results of other turbulence models together with experimental results.

In *large eddy simulation (LES)*, only the dissipative motions, the micro-structures, are modeled and the rest of the motions are resolved. It simulates the problem with a reasonable accuracy, which is comparable to the accuracy of DNS with less computational efforts. LES can be used for flows having the effect of irrotational strains and normal stress due to being anisotropic.

The grid-point requirements for DNS of single phase channel flow, N_{DNS}, is [5]

$$N_{DNS} \sim (3Re_\tau)^{9/4}. \tag{1.1}$$

where $Re_\tau = u^{'}L/\nu$ is the turbulent Reynolds number and $u^{'}$ is the root mean square (RMS) of the velocity. Even for the single phase case that has relatively low Reynolds number, the computational cost of DNS is high.

The grid-point requirements for LES of single phase channel flow can be estimated with respect to the requirement of DNS [5]

$$N_{LES} \approx \left(\frac{0.4}{Re_\tau^{1/4}}\right) N_{DNS}. \tag{1.2}$$

The number of grid points required for numerical simulation varies with the use of the wall-modeled and wall-resolved LES. These different approaches of LES are discussed in more detail in Section 2.2.

In the experiment that has been carried out in the Laboratory for Aero- and Hydrodynamics of Faculty of Mechanical, Maritime and Materials Engineering of Delft University of Technology, a circular pipe, which has a diameter of 0.05 m and a length of 10 m, has two fluids flowing at ambient pressure and temperature. Each of the phases flow through the pipe with a different viscosity and density. The flow characteristics are determined by

the shear stresses and gravity, which are affecting the interface and flow near the walls. When the flow rate is at a moderate level, the effect of gravity is observed on the flow, i.e. the stratification occurs and the phase with the higher density flow through the bottom region and the phase with the lower density flow through the top region. Both fluids are assumed to be incompressible and separated by an interface. The flow becomes fully developed over a length of 7.5 m.

There can be fluctuations at the interface between the two phases when the gas flow rate increases, despite the fact that the flow of the liquid phase remains laminar. For air and water this occurs when the air phase becomes turbulent; $Re_{air} \approx 3500$. The water phase is turbulent when $Re_{water} \approx 3400$. The Reynolds number Re is defined as $Re_f = u_f D_{fh}/\nu_f$, where u_f is the bulk velocity of the fluid, D_{fh} is the hydraulic diameter, and ν_f is the kinematic viscosity of the fluid. In air phase, $D_{gh} = 4A_g/(S_{wg} + S_{int})$, and in water phase $D_{lh} = 4A_l/S_{wl}$, where A_g and A_l are the cross-sectional areas respectively for gas and liquid phases, and S_{wg}, S_{wl}, and S_{int} are the wetted perimeters [3]. For the turbulent non-wavy stratified case (intermittency factor ∼0.99; which is the fraction of time that motion is turbulent) the Reynolds number is ∼3400. The friction Reynolds number [4] for the single phase pipe flow is $Re_\tau = 395$ [6]. Most importantly, turbulence can be carried over to the other phase through the interface. If the transition occurs from stratified flow to stratified-wavy flow, the interfacial momentum transfer varies due to the existence of waves at the interface. This process makes modeling of the momentum transfer complicated. Therefore, in order to make the problem slightly easier two-phase flow is considered to be modeled with a single-phase turbulence model while ignoring the momentum transfer and concentrating on modeling the turbulence in both phases away from the interface.

1.3 Research objective

The objective of this study is to analyze different turbulence models by getting more insight into the current state-of-the-art of modeling turbulent two-phase stratified flow. The results of the experiment that has been carried out in the Laboratory for Aero- and Hydrodynamics of Faculty of Mechanical, Maritime and Materials Engineering of Delft University of Technology [6] are intended to be used to validate the results of this study together with the results of Eggels [1]. Yet due to time constraints two-phase turbulent stratified flow is not realized. Therefore, the renewed aim of this thesis is to simulate the pipe flow discussed in the work of Eggels [1] and compare the resulting velocity profiles. The main difficulty to resolve such a flow is the feasibility and the complexity of performing such a simulation using either DNS, LES or another turbulence model for both turbulent single phase and two-phase flows.

The mimetic discretization of the Navier-Stokes equations are intended to be used together with an extended diffusive transport term to include variable viscosity. However, because of the time available the standard formulation is used for DNS which assumes a constant

[3] The wetted perimeter is the length of the total surface in contact with the fluid. For a single-phase pipe flow, the wetted perimeter is equal to πD, where D is the diameter of the pipe.

[4] It is defined as the ratio of inner (close to the wall) and outer length scales (further away from the wall), $Re_\tau = u^*\delta/\nu = \delta/\delta_\nu$, where u^* and δ_ν are defined in eqs. (2.37) and (2.38) respectively, and δ represents the outer layer length scale for the flow.

viscosity. Thus, the algorithm the formulation has to be extended to allow for a variable viscosity (subgrid scale model). Otherwise, it is not possible to simulate the problem using LES method. Furthermore, a boundary condition that allows to solve for the pressure difference over the pipe length as part of the problem needs to be implemented to the algorithm. For that reason, a flow rate is to be specified with respect to the bulk (mean) velocity, inflow and outflow pressures.

In this thesis, LES is found to be applicable from a computational point of view. However, it is not known how to model the crucial momentum transfer between gas and liquid phases. In order to validate this work, the results of this study should be as close as possible to the results of the experiment that includes the effect of momentum transfer between the phases and to the turbulent single-phase pipe flow results of [1]. For this reason, the computational cost and accuracy of DNS and LES in single-phase flow are analyzed. Then, necessary periodic boundary conditions for the pressure and velocity are introduced. In the next and final phase, single-phase pipe flows are simulated using DNS and LES. The single phase turbulent flow is simulated with a variable viscosity formulation using the subgrid scale model.

1.4 Thesis outline

Turbulence models for incompressible single-phase flows are discussed in Section 2 in order to compare the computational cost and accuracy of different models. This way an insight is obtained about the number of unknowns required to model the two-phase flow, which is similar to the single-phase case in terms of wall and inlet-outlet boundary conditions and the inner region flow regime. In Section 2.2, the LES method is elucidated together with other necessary models such as the Smagorinsky model to resolve all turbulence scale structures. The minimum computational cost that is possible with the LES method for single-phase flows is calculated to be able to make inferences about the two-phase case in Section 2.3.1, and to decide how to proceed with the turbulent two-phase flow modeling. In particular, DNS and LES methods are identified as the appropriate techniques in the literature review and investigated carefully to describe turbulent stratified two-phase flows, especially in the near-wall region and at the interface in order to realize and decrease the computational complexity of the problem.

In Chapter 3, the physics of stratified two-phase flow is investigated. The equations for two-phase flow are identical to the single-phase case when the effects of surface tension can be neglected, the only differences between two-phase flow and single phase flow equations are the variable density and viscosity. Hence, as an initial approximation the influence of the interface can be ignored (both its influence on the mixing length and on the momentum transfer between both phases) and a simple single-phase flow model can be used. In that manner, the variable viscosity formulation is adapted to the present code but density is not modified to allow for a variable formulation. Therefore, the aim to model turbulent stratified two-phase flows while ignoring the effect of the interface is not achieved during the thesis period. Dimensionless parameters that are commonly used in two-phase flows are explained and the ones that are relevant to this study are clearly stated in Section 3.1. The governing equations and boundary conditions for two-phase flows are discussed in Section 3.2 while neglecting the effect of the interface and considering each phase on its

own as a single-phase flow to simplify the problem. Several computational techniques for two-phase turbulent flow are described in Chapter 4. In Section 4.1 different turbulence models that have been used in the literature are investigated.

In order to get a better insight about turbulent stratified two-phase flows the influence of the interface and the momentum transfer in between two phases needs to be considered and modeled appropriately. Although, this is not the main scope of this research (but can be for future researches), the influence of the interface and the one-fluid model are explained in Appendix E. Specific version of the *Mass-Conserving Level Set (MCLS)* method, which is developed for the discretization of three-dimensional equations that describes immiscible incompressible two-phase flow in a circular pipe geometry is elucidated in Appendix E.1. The mimetic discretization method described in Section E.1.2 is intended to be used but is not utilized since turbulent two-phase flows are not modeled in this thesis.

Implementation of periodic and no-slip boundary conditions are given in Chapter 5. Several test cases concerning the Poiseuille flow in 2D and 3D are developed and the results obtained from `MATLAB` are compared with the results of the present algorithm for laminar flow. The algorithm is modified such that it allows for a variable viscosity. The variable viscosity formulation both in Cartesian and cylindrical coordinates are implemented In Chapter 6. The variable viscosity assumption is validated by comparing the analytical velocity profiles with the numerical ones. Preconditioned `GMRES` and its efficiency for both momentum and pressure equations are given in Section 6.4. The algorithm that solves for the pressure difference over the pipe length as part of the problem is validated for a variable viscosity formulation in Section 6.5.

Numerical results of DNS and LES computations for turbulent single-phase pipe flows are presented in Chapter 7. The aim of these computations is to clarify the performances of the numerical techniques (via DNS) and the SGS modeling (via LES). Descriptions and results of DNS and LES computations are given in Sections 7.1.1, 7.1.2, 7.2.1 and 7.2.2 respectively. The report ends with Section 8 with conclusions and recommendations for further research.

Chapter 2

Turbulence modeling for single-phase flows

In this chapter, different turbulence models for single-phase flow are discussed and compared. The background of *Direct Numerical Simulation (DNS)*, *Reynolds averaged Navier-Stokes (RANS)*, and *Large Eddy Simulation (LES)* are presented in detail. In this thesis, Reynolds averaging models are not considered due to insufficient accuracy. Only DNS and LES are analyzed elaborately for turbulent single-phase pipe flows. Since all details of the flow are resolved in DNS, this approach is not feasible for high Reynolds number flows. DNS is a useful approach to get more insight into the physical processes involved in the problem. In LES, large scales of turbulent motions are resolved and the small scales (subgrid scales) are modeled accordingly. For flows with large fluctuations, the LES technique is expected to be more reliable and accurate than Reynolds averaged models (e.g. flow over bluff bodies, which has unsteady separation and vortex shedding [2]). The computational cost of LES is smaller than DNS but larger than Reynolds averaged models. At the end of each section, the properties of each individual method is summarized and its limitations are discussed and compared with the other methods.

2.1 Introduction

Turbulent flow is three dimensional, chaotic, diffusive, quasi-random, dissipative and intermittent. In turbulent flow, the field parameters (e.g. velocity and pressure) are not deterministic, but random functions of space and time and are characterized by random fluctuations in all directions. The tensor notation (in particular, the Einstein summation convention [1]) of the conservation equation of mass for an incompressible fluid with

[1] In Einstein summation convention, the repeated occurrence of the same subscript in a single term indicates that these terms should be summed over all possible values of the repeated subscript. It is used to simplify expressions including summations of vectors, matrices, and tensors.

constant viscosity is:

$$\frac{\partial u_i}{\partial x_i} = 0, \tag{2.1}$$

where u and p are velocity and pressure fields respectively. The flow is governed by incompressible Navier-Stokes equations and the conservation equation for momentum is:

$$\rho\frac{\partial u_i}{\partial t} + \rho\frac{\partial}{\partial x_j}(u_j u_i) = -\frac{\partial p}{\partial x_i} + \frac{\partial}{\partial x_j}(2\mu S_{ij}) + g_i, \tag{2.2}$$

where ρ, μ, S_{ij} and g are density, viscosity, strain-rate tensor and gravity respectively. The strain-rate tensor is as follows:

$$s_{ij} = \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right). \tag{2.3}$$

Together with the continuity Eq. (2.1), the equations of motion can be written as:

$$\rho\frac{\partial u_i}{\partial t} + \rho u_j\frac{\partial u_i}{\partial x_j} = -\frac{\partial p}{\partial x_i} + \mu\frac{\partial^2 u_i}{\partial x_i \partial x_j} + g_i. \tag{2.4}$$

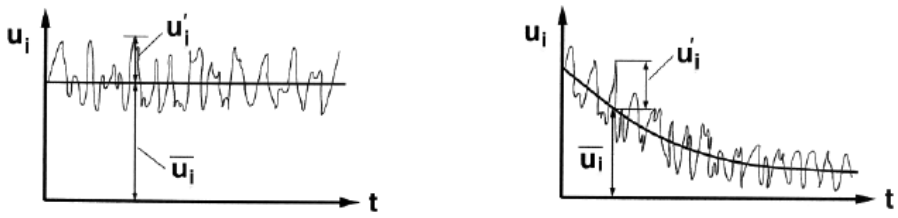
The Navier-Stokes equations are non-linear and difficult to solve analytically. The exact solution can only be obtained for simple flow configurations, which are only realistic in the simple configurations they describe. Therefore, it is hard to get more insight into the nature of turbulence by analytically solving these equations.

Due to the large computational resources required to resolve the flow at the appropriate length and time-scale, an additional modeling is required to avoid having to solve for the small temporal and spatial time scales. The need to model additional equations for the new unknown terms is called *Turbulence Modeling*. These are the turbulence models based on Reynolds Averaged Navier-Stokes (RANS) equations (time averaged) in the order of increasing complexity:

- Algebraic (zero equation) models: mixing length (first order model),
- One equation models: k-model, ν_t-model (first order model),
- Two equation models: $k-\epsilon$, $k-\omega^2$ (first order model),
- Algebraic stress models (second order model),
- Reynolds stress models (second order model).

The flow is considered to be statistically stationary in order to be able to use RANS models for solving turbulent flows, which means that the joint probability distribution (i.e. the likelihood of occurrence of two events at the same time and together [2]) of the flow does not change when time is shifted. As a result, the mean and variance of the flow parameters are constant over time and do not have a pattern. By this means, the velocity field u_i and the pressure field p can be decomposed into a mean (time-averaged) and fluctuating part:

$$u_i = \overline{u_i} + u_i^{'}, \quad p = \overline{p} + p^{'}. \tag{2.5}$$



(a) Statistically stationary (b) Statistically unsteady

Figure 2.1: Statistically stationary flows, where u_i:=instantaneous velocity, $\bar{u}_i$:=mean velocity (time-averaged velocity), and $u_i^{'} := u_i - \overline{u}_i$:=velocity fluctuation

The aim is to obtain set of equations to describe the average properties of the turbulent flow. The time average is defined as:

$$\overline{f} = \lim_{T\to\infty} \frac{1}{T} \int_{t_0}^{t_0+T} f \, dt. \tag{2.6}$$

Introducing the decomposition (2.5):

$$\rho \left[\frac{\partial(\overline{u_i} + u_i^{'})}{\partial t} + (\overline{u_j} + u_j^{'}) \frac{\partial(\overline{u_i} + u_i^{'})}{\partial x_j} \right] = -\frac{\partial(\overline{p} + p^{'})}{\partial x_i} + \mu \frac{\partial^2(\overline{u_i} + u_i^{'})}{\partial x_j \partial x_j}, \tag{2.7}$$

$$\frac{\partial(\overline{u_i} + u_i^{'})}{\partial x_i} = 0. \tag{2.8}$$

Applying the decomposition (assuming that the flow is statistically stationary, Fig. 2.1a) and the rules of averaging, the following *Reynolds Averaged Navier-Stokes (RANS)* equations are obtained:

$$\rho \left[\frac{\partial \overline{u}_i}{\partial t} + \overline{u_j} \frac{\partial \overline{u}_i}{\partial x_j} \right] = -\frac{\partial \overline{p}}{\partial x_i} + \frac{\partial}{\partial x_j} \left(\mu \frac{\partial \overline{u}_i}{\partial x_j} - \rho \overline{u_i^{'} u_j^{'}} \right), \tag{2.9}$$

$$\frac{\partial(\overline{u_i} + u_i^{'})}{\partial x_i} = 0. \tag{2.10}$$

However, application of the Reynolds decomposition leads to new unknowns, which are called Reynolds stresses and turbulent fluxes. The Reynolds stress tensor is defined as:

$$\tau_{ij} := \rho \overline{u_i^{'} u_j^{'}}. \tag{2.11}$$

In Newtonian fluids, the molecular shear stress is given by:

$$\tau_{ij}^{mol} = -\mu_T \left[\frac{\partial \overline{u_i}}{\partial x_j} + \frac{\partial \overline{u_j}}{\partial x_i} \right] = -2\mu \overline{S_{ij}}, \tag{2.12}$$

where $\overline{S_{ij}}$ is the mean-rate of strain tensor. By using the turbulent-viscosity (also known as Boussinesq) hypothesis, the deviatoric (anisotropic) Reynolds stress is proportional to the mean rate of strain [7]:

$$\tau_{ij}^{RANS,a} = \rho \overline{u_i^{'} u_j^{'}} - \frac{2}{3} \rho k \delta_{ij} = -\mu_T \left[\frac{\partial \overline{u_i}}{\partial x_j} + \frac{\partial \overline{u_j}}{\partial x_i} \right] = -2\rho \nu_T \overline{S_{ij}}, \tag{2.13}$$

[2]The probability of event x occurring at the same time with event y.

where a in the superscript refers to the anisotropic part of Reynolds stress, and $\nu_T(\vec{x},t)$ is the turbulent viscosity. Thus, the momentum equation is as follows:

$$\frac{\partial \overline{u}_i}{\partial t} + \overline{u}_j \frac{\partial \overline{u}_i}{\partial x_j} = -\frac{1}{\rho}\frac{\partial}{\partial x_i}(\overline{p} + \frac{2}{3}\rho k) + \frac{\partial}{\partial x_j}\left[\nu_{eff}\left(\frac{\partial \overline{u}_i}{\partial x_j} + \frac{\partial \overline{u}_j}{\partial x_i}\right)\right], \tag{2.14}$$

where $\nu_{eff} = \nu + \nu_T(\vec{x},t)$. By specifying $\nu_T(\vec{x},t)$, which is not a constant (since a constant do not change the same equation with the same unknowns), the closure problem is solved (instead of $\overline{p}$ and k, $q = \overline{p} + \frac{2}{3}\rho k$ is only used).

The reason for not considering Reynolds averaged models in this thesis is the large number of model equations. The extensive modeling in RANS yields unpredictable results. Therefore, it not considered in the rest of the thesis as a promising turbulence model for solving turbulent two-phase flows.

2.2 Large eddy simulation (LES)

The turbulence model LES lies in between RANS and DNS in terms of computational cost. The DNS method consumes a considerable part of the computational resources for resolving the dissipative range, whereas most of the energy and anisotropy is contained in the large scales. RANS methods model all the turbulence spectrum, and results are in agreement with experiments at very high Reynolds number. However, due to quite strong assumptions in RANS models, the results cannot be predicted *a priori*. Thus, LES technique is used to resolve the large scale three dimensional unsteady turbulent motions of the flow explicitly while the interactions in the small scales (dissipative range) are modeled.

Large Scales	**Small Scales**
Produced by mean flow	Produced by large scales
Inhomogeneous, anisotropic	Homogeneous, isotropic
High energy, long life	Low energy, short life
Diffusive	Dissipative
→ Difficult to model: Universal model not possible	→ Simple to model: Universal model may work

Table 2.1: Comparison of properties of turbulence scales

The LES modeling consists of four parts:

1. Filtering operation: Decompose velocity field in large (resolved) and small scales (SGS:= sub-grid scale); $U(x,t) = \overline{U}(x,t) + u^{'}(x,t)$, $\overline{U}(x,t)$ is three dimensional and time dependent and represents the large eddies,

2. The filtered velocity is described with the Navier-Stokes equations and SGS stress tensor,

3. Closure is provided by using the SGS stress tensor model (eddy viscosity model),
4. The filtered equations are solved for the velocity and pressure fields.

The grid size in LES is usually smaller than RANS based models (except in the boundary layer where the grid sizes are comparable) in order to resolve small spatial scales. The numerical approximation will continue to converge to the exact solution of the model upon grid refinement (i.e. the solution of the continuous equations depends on the mesh width, as anything that is not resolved on the mesh has to be modeled). In LES models, results have a grid dependency, the smaller the sizes of the grid the better the accuracy of the result. When the mesh is very fine, the result can converge to the result of DNS, in which the flow is fully resolved instead of modeled like in RANS.

2.2.1 Filtering

The separation of small and large scales is achieved by applying low pass filtering. Afterwards, the filtered velocity field can be resolved on a relatively coarse grid where the necessary grid spacing Δx is defined proportional to the filter width Δ (e.g. $\Delta x = 0.5\Delta$ in contrast to $\Delta x = 2\eta$ in DNS, where η is the Kolmogorov length scale). The ideal choice for Δ can be shown to be $\Delta < l_{EI}$ with l_{EI} being the size of the smallest energy motions. Large eddies (coarse structures) are larger than Δ and small eddies (fine structures) are

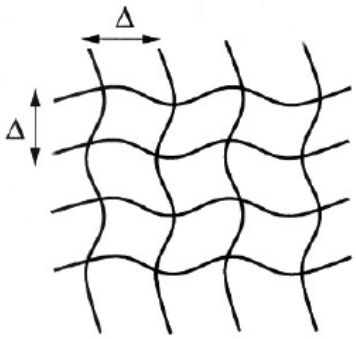


Figure 2.2: Demonstration of the filtering operation

smaller than Δ. At a given point x in the computational domain, the filtering operation is expressed as

$$\overline{U}(x,t) = \int G(r,x)U(x-r,t)\ dr, \tag{2.15}$$

where G is the specified filter function, and the integral is all over the entire flow domain. The filter function satisfies

$$\int G(r,x)\ dr = 1. \tag{2.16}$$

The simplest filter is the homogeneous filter: $G(r,x) = G(r)$ (at every point the same filter is applied). Gaussian, box, and spectral filters are the most commonly used filters [7]. The residual velocity field is defined by

$$u^{'}(x,t) = U(x,t) - \overline{U}(x,t). \tag{2.17}$$

Thus, the velocity field is decomposed similarly as the Reynolds decomposition:

$$U(x,t) = \overline{U}(x,t) + u^{'}(x,t), \tag{2.18}$$

and $u^{'}(x,t)$ is time dependent. Hence the filtered residual is not zero: $u^{'}(x,t) \neq 0$. The filtered velocity can be expressed by a convolution in one dimension:

$$\overline{U}(x) = \int_{-\infty}^{\infty} G(r)U(x-r)\ dr. \tag{2.19}$$

Conservation equations need to be formulated for the filtered velocity field (homogeneous filter, $G(r)$ is considered). In order to obtain the equations, the filtering operation is applied to the Navier-Stokes equations. The filtered continuity equation is

$$\left(\overline{\frac{\partial U_i}{\partial x_i}}\right) = \frac{\partial \overline{U}_i}{\partial x_i} = 0. \tag{2.20}$$

Thereby, under the assumption that filtering and differentiation commute, divergence of the residual velocity field is

$$\frac{\partial u_i^{'}}{\partial x_i} = \frac{\partial}{\partial x_i}(U_i - \overline{U}_i) = 0. \tag{2.21}$$

For the momentum equation the filtering operation results in the following equation:

$$\frac{\partial \overline{U}_j}{\partial t} + \frac{\partial \overline{U_i U_j}}{\partial x_i} = \nu \frac{\partial^2 \overline{U}_j}{\partial x_i \partial x_i} - \frac{1}{\rho}\frac{\partial \overline{p}}{\partial x_j}, \tag{2.22}$$

where $\overline{p}(x,t)$ is the filtered pressure field. The residual stress tensor is introduced in order to make this equation similar to the Navier-Stokes equation ($\overline{U_i U_i} \neq \overline{u}_i \overline{u}_j$):

$$\tau_{ij}^R = \overline{U_i U_j} - \overline{U}_i\,\overline{U}_j, \tag{2.23}$$

$$\overline{U_i U_j} = \tau_{ij}^R + \overline{U}_i\,\overline{U}_j. \tag{2.24}$$

The residual kinetic energy is

$$k_r = \frac{1}{2}\tau_{ij}^R, \tag{2.25}$$

and the anisotropic residual stress tensor is defined as

$$\tau_{ij}^r = \tau_{ij}^R - \frac{2}{3}k_r \delta_{ij}. \tag{2.26}$$

When the filtered pressure is expressed as $\overline{p} = \overline{p} + 2/3\rho k_r$, the isotropic part of the residual stress is obtained. The modified filtered momentum equation is

$$\frac{\partial \overline{U}_j}{\partial t} + \overline{U}_i \frac{\partial \overline{U}_j}{\partial x_i} = \nu \frac{\partial^2 \overline{U}_j}{\partial x_i \partial x_i} - \frac{\partial \tau_{ij}^r}{\partial x_i} - \frac{1}{\rho}\frac{\partial \overline{p}}{\partial x_j}. \tag{2.27}$$

The filtered Eq. (2.27) is not closed as it was the case in $k-\epsilon$ model and RSM. Thus, the equation should be closed by modeling the residual stress tensor τ_{ij}^r. The residual stress tensor introduces additional dissipation: It removes energy from large scales and the energy is transferred to the smaller scales. The filtered velocity $\overline{U}_i$ depends on the filter (filter type and width) indirectly through the model for τ_{ij}^r.

The convective flux is defined as

$$\overline{U_i U_j} := \overline{U_i}\,\overline{U_j} + \tau_{ij}^R = \overline{U_i}\,\overline{U_j} + \tau_{ij}^r + \frac{2}{3}k^r \delta_{ij}, \tag{2.28}$$

where the decomposition of the residual stress is

$$\tau_{ij}^R := L_{ij} + C_{ij} + R_{ij}, \tag{2.29}$$

$$L_{ij} := \overline{\overline{U_i}\,\overline{U_j}} - \overline{U_i}\,\overline{U_j}, \tag{2.30}$$

$$C_{ij} := \overline{\overline{U_i} u_j'} + \overline{u_i' \overline{U_j}}, \tag{2.31}$$

$$R_{ij} := \overline{u_i' u_j'}. \tag{2.32}$$

The tensors L_{ij}, C_{ij}, and R_{ij} are called the Leonard stresses, the cross stresses, and the SGS Reynolds stresses respectively [5].

2.2.2 The Smagorinsky model

The anisotropic residual stress tensor can be modeled and the system of equations can be closed with one of the simplest model called *the Smagorinsky model* as follows:

$$\tau_{ij}^r = -2\nu_t \overline{\mathcal{S}}_{ij}, \tag{2.33}$$

where $|\overline{\mathcal{S}}| := \sqrt{2\overline{S}_{ij}\,\overline{S}_{ij}}$ is the characteristic filtered rate of strain, $\nu_r = l_S^2 \overline{S}_{ij}$ is the eddy viscosity (or the turbulent viscosity) of the residual motions, and $\overline{S}_{ij} := \frac{1}{2}\left(\frac{\partial \overline{U}_i}{\partial x_j} + \frac{\partial \overline{U}_j}{\partial x_i}\right)$ is the filtered rate of strain tensor.

The model for eddy viscosity can be expressed as:

$$\nu_t = l_S . l_S \overline{S}, \tag{2.34}$$

where l_S is the Smagorinsky length scale, which only affects the small scales and is proportional to the filter width, i.e. $l_S = C_S \Delta$, and C_S is the Smagorinsky coefficient. The second part $l_S \overline{S}$ expresses the velocity part.

The rate of transfer of the energy to the residual motion is

$$P_r = -\tau_{ij}^r \overline{S}_{ij} = 2\nu_t \overline{S}_{ij} \overline{S}_{ij} = \nu_t \overline{S}^2 > 0, \tag{2.35}$$

which means energy is always removed since $\nu_t > 0$ (i.e. the energy is transferred only from filtered to the residual motion). The LES equation itself does not depend on the chosen filter. The filter only affects $-\partial \tau_{ij}^r / \partial x_i$. The mean energy transfer is balanced by the dissipation for high Reynolds number flows:

$$\epsilon = \overline{P}_r = \overline{\nu_t \overline{S}^2} = l_S^2 \overline{\overline{S}^3}. \tag{2.36}$$

The simple Smagorinsky model can be improved such that inhomogeneous turbulence can also be modeled. The value of the Smagorinsky coefficient is equal to zero (i.e. $C_S = 0$) close to the wall and also for laminar flows. Moreover, the Smagorinsky constant

has a different value for different types of flows (e.g. for high Reynolds number flows $C_S \approx 0.15$). Because of all these reasons, a more advanced model is needed to specify a general value for C_S (refer to Appendix D for more information about the dynamic Smagorinsky model). Also, if non-uniform grid spacing is going to be used in radial direction, then the conventional Smagorinsky model becomes less accurate in the near-wall region. Therefore, it is preferable to use a more advanced and *'universal'* model to obtain better statistical results.

2.2.3 Wall treatment

Accurately modeling viscous effects in the near-wall region is an important challenge in CFD. These viscous effects in the near-wall regions need to be modeled using appropriate wall functions which will be discussed later.

In the presence of a solid wall, vorticity is generated and a turbulent boundary layer will occur. Close to the wall, the wall shear stress τ_w and the viscosity ν play an important role. This region is called the *viscous sublayer*, whereas the outer region, where large scale turbulent eddy shear forces dominates, is called the *outer layer*. In between these two layers, there exists an overlap layer called the *log-law region* in which the velocity profile shows a logarithmic variation (see Fig. 2.3 and Fig. 2.4). In the viscous sublayer region, the effects of the pressure gradient and convection are assumed to be negligible. The important parameters in that region are density, viscosity, wall shear stress and normal distance from the wall. On the other hand, in the outer region, where the convection and pressure gradient are dominant, the effect of viscosity is assumed to be negligible.

In the near-wall region new parameters are defined that are called the *viscous scales*. A reference length scale (viscous length scale) and velocity scale (friction velocity) are defined as follows:

$$u^* = \sqrt{\frac{\tau_w}{\rho}}, \tag{2.37}$$

and the viscous length scale:

$$\delta = \nu \sqrt{\frac{\rho}{\tau_w}} = \frac{\nu}{u^*}. \tag{2.38}$$

These can be used to define a dimensionless velocity and a dimensionless length (also called wall unit) as:

$$u^+ := \frac{u}{u^*}, \tag{2.39}$$

$$y^+ := \frac{y}{\delta} = \frac{u^* y}{\nu}, \tag{2.40}$$

where u is the velocity component parallel to the wall, y is the distance normal to the wall, u^* is the shear stress velocity and ν is the kinematic viscosity of the fluid. The law of the wall defines the average dimensionless velocity of a turbulent flow which is proportional to the logarithm of the distance (dimensionless length or wall unit) from a certain point to the wall (Fig. 2.4). The first layer is called the viscous sublayer (see Fig. 2.4) and at high Reynolds number the viscous sublayer is very thin ($y^+ < 5$). Thus, special near-wall treatments need to be applied. There are two possibilities for turbulence models in order to resolve the flow in the near-wall region; the low Reynolds number method, in which

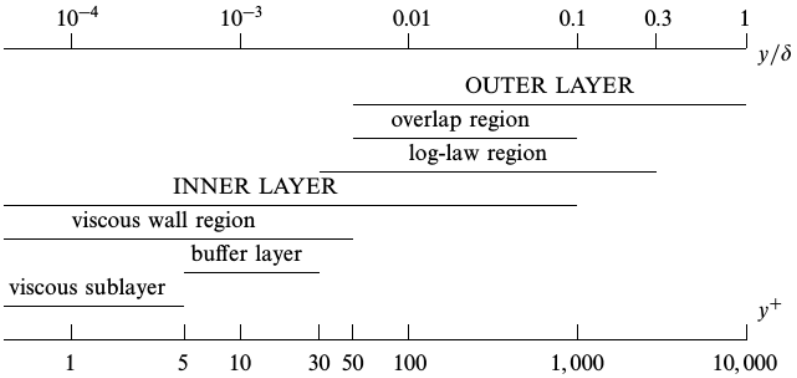


Figure 2.3: The law of the wall: layers defined in terms of y/δ for turbulent channel flow at $Re_\tau = 10^4$ [2]

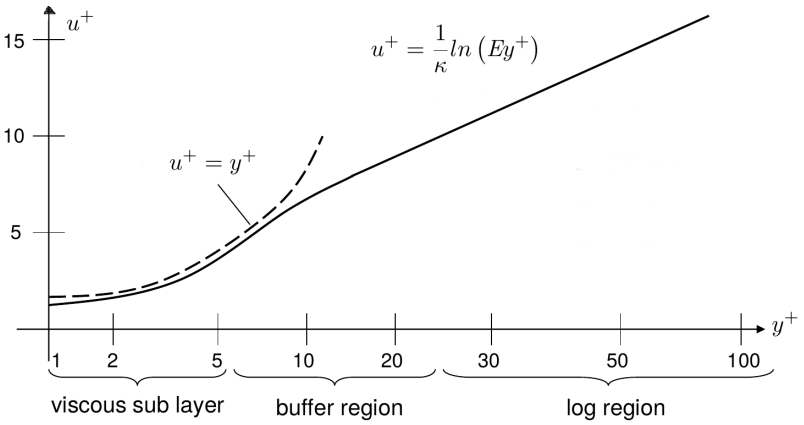


Figure 2.4: Near-wall mean velocity profiles, wall regions and layers

the mesh is very fine close to the wall, and the high Reynolds number method, in which *wall functions* are implemented. The second approach is computationally less intensive. However, important information about the physics of the problem is lost. Over some region of the wall layer, viscous effects are large due to the no-slip boundary condition at the wall. In low Reynolds number, a way to overcome these problems is to introduce damping effects. Another option is to use wall functions, in which the flow in the near-wall region is modeled. By using wall functions, empirical laws are provided such that these laws make it possible to express the mean velocity parallel to the wall and turbulence parameters. Wall functions, which are based on the law of the wall and valid only in the log region, provide boundary conditions for the momentum and turbulent transport equations near the wall, instead of conditions at the wall itself. As a result, the viscous sublayer does not have to be resolved and the fluctuating flow parameters near the wall can be resolved without using a very fine mesh.

There are two specific approaches for wall treatment in LES, LES with near-wall resolution (LES-NWR) and LES with near-wall modeling (LES-NWM). In LES-NWR, the flow is resolved everywhere up to 80% of the energy, also taking into account the energy in the

viscous layer (the filter and grid spacing are fine enough). However, the flow is not resolved in near-wall region in LES-NWM. Thus, 80% of the energy is not obtained in the viscous layer.

The production, dissipation, kinetic energy, and Reynolds stress anisotropy reach their maximum values in the viscous layer (at $y^+ < 20$) [2]. The filter width should be of the same order of the viscous length scale δ in order to resolve the viscous sublayer in the near-wall region with LES-NWR. Therefore, the number of grid points required increases drastically, proportional to $Re^{1.76}$ [2]. Therefore, LES-NWR is not practical for very high Reynolds number flows considering the available computational resources. On the other hand, LES-NWM is independent of the Reynolds number since the grid spacing and the filter width are proportional to the flow length scale l. The reason for that the flow is modeled in the near-wall region instead of resolved.

2.2.4 Wall functions

It is important to model the flow in the near-wall region accurately because walls are the main source of vorticity and turbulence. Thus, in order to get an accurate result, wall functions or some other method should be used. In wall functions, the first point of the grid is assumed to be in the logarithmic layer and called y_p, which is an artificial parameter. The accuracy of the result depends on the choice of y_p. If the first grid-point is too close (i.e. located in the linear sublayer $y^+ < 5$), then the dimensionless velocity is equal to dimensionless length [7]:

$$u^+ = y^+. \tag{2.41}$$

If the mean flow is parallel to the wall, then the log-law relations apply (log-law region: $y^+ > 30$) and the law of the wall for mean velocity yields;

$$u^+ = \frac{1}{\kappa} ln\left(Ey^+\right), \tag{2.42}$$

where u^+ is the dimensionless velocity, κ is the von Kármán constant ≈ 0.41, E is an empirical constant $= 9.793$, and y^+ is the dimensionless length [7].

In the buffer layer, $5 < y^+ < 30$, none of the laws hold. Therefore, when $y^+ < 11$, linear approximation is more accurate, and when $y^+ > 11$, the logarithmic approximation is more accurate [8].

A no slip condition is imposed at the wall (i.e. $u \neq 0$). In order to set boundary conditions for k and ϵ at the grid-point adjacent to the wall, the friction velocity and the wall shear stress should be computed. Substituting eqs. (2.40) and (2.41) into eqs. (2.37) and (2.38) yields [2]:

$$\frac{u}{u^*} = \frac{1}{\kappa} ln\left(E\frac{u^* y}{\nu}\right), \tag{2.43}$$

for the log-law region, and

$$\frac{u}{u^*} = \frac{u^* y}{\nu}, \tag{2.44}$$

for the viscous sublayer.

In the log region, production and dissipation of turbulent kinetic energy are almost equal $P \approx \rho\epsilon$. The production term for a simple two dimensional boundary layer, when the $y = x_2$ direction is normal to the wall, is $P := -\tau^t du/dy$. This results in the following:

$$-\tau^t \frac{du}{dy} = \rho\epsilon. \tag{2.45}$$

The turbulent shear stress can also be expressed as:

$$\tau^t = \mu_t \frac{du}{dy} = C_\mu \rho \frac{k^2}{\epsilon} \frac{du}{dy}. \tag{2.46}$$

Solving Eq. (2.45) for ϵ using eqs. (2.39) and (2.44) yields the turbulent kinetic energy:

$$k = \frac{(u^*)^2}{\sqrt{C_\mu}}, \tag{2.47}$$

where u_τ^* is the friction velocity and the dissipation is

$$\epsilon_p = \frac{(u^*)^3}{\kappa y_p}. \tag{2.48}$$

2.2.5 Overview

The smallest finite difference cells in LES can be larger than the Kolmogorov length scale. Therefore, larger time steps can be taken compared to DNS since all flow scales smaller than filter size will be modeled. This leads to less computational effort (in terms of memory and CPU) than DNS since it also resolves the smallest eddies.

In order to decrease the total time to solve the problem with LES, wall functions can be imposed as a boundary condition, which will reduce the resolution requirements. If the law of the wall is used in the viscous sublayer, then the number of grid points decreases. However, using the law of the wall as a boundary condition cannot predict fluctuating values in the log-law region. Thus, the law of the wall may not be sufficient to see the changes of the kinetic energy and dissipation in LES [5]. On the other hand, it is relatively expensive to resolve the near-wall region at high Reynolds number flows. Hence, wall stress models can be used to provide necessary wall stresses to the LES. In this way, the computational cost will significantly decrease with the use of the model.

For free shear flows, although, the small-scale turbulence in the initial part of the shear layer is not adequately resolved, LES is still a good choice since the computational cost of LES is independent of the Reynolds number. For pipe flows, LES is not quite practicable since the motions that contain energy near the wall are challenging to resolve. However, for this difficulty LES-NWM can be utilized to model the near-wall region and decrease the cost of the problem.

2.3 Estimating problem complexity for different turbulence models

Modeling multiphase turbulent flows with high accuracy is more difficult than modeling single-phase turbulent flows. Although, the interface between two immiscible phases can be described quite precisely with available models, the influence of the turbulent fluctuations in one of the phases may have great influences on the dynamics of the interface. Therefore, it is very important to identify the effect of the turbulence in all phases.

There are some studies about turbulent two-phase stratified pipe flow but not much of them have been carried out with LES since it is not a very common practice in turbulent two-phase stratified pipe flow. Therefore, in order to have an idea about the flow properties, first, DNS and LES are used for turbulent single-phase pipe flows, then, necessary inferences are made about turbulent two-phase pipe flows. However, the required number of unknowns for RANS equations are not estimated since the limitations (which are almost negligible compared to other techniques) imposed by RANS based models are quite less than the limitations imposed by DNS and LES. Also, the Reynolds number is too low for doing RANS. By estimating the number of total grid points required for single-phase turbulent flow, a minimum requirement and an insight about the complexity of turbulent two-phase flow is obtained.

2.3.1 Computational costs

As an initial ambition the experiments that have been carried out by [6] are planned to be used for validation purposes. In these experiments, the diameter of the pipe is $D = 50\ mm$, and the length of the pipe is $L_x = 200D = 10\ m$. The superficial water flow rates are $0.0085\ m/s$ and $0.0255\ m/s$, and the superficial air velocity varies from 0 to $5.4\ m/s$ (non-wavy surface between 0 and $1\ m/s$) to obtain both laminar and turbulent flow respectively.

As the velocity of the air varies in the experiments, the velocity of the liquid also varies due to the momentum transfer between the phases at the interface. When the air velocity is $u_G = 1.04\ m/s$, the gas phase is turbulent, and the liquid phase becomes (stratified) turbulent at a liquid velocity $u_L = 0.1126\ m/s$ with an intermittency factor 0.99 and a Reynolds number $Re_L = u_L D/\nu = 3421$. The Reynolds number for the air at that velocity is $Re_G = 3632$ [6]. With an increase in the air velocity, the interface becomes oscillatory without showing regular wave patterns. This is caused by the turbulence in the gas phase. Around a gas velocity of $3.5\ m/s$ waves start to appear on the interface. The transition from smooth interface to wavy interface also depends on the velocity of the liquid phase.

The length of the problem domain that is resolved should be long enough to accommodate the largest turbulence structures. In channel flow, eddies are stretched parallel to the channel walls, and their length is approximately equal to $2H$, where H is the height of the channel [5]. For pipe flow, in order to compute the required pipe length for an accurate model that includes the largest turbulent structures, the two-point correlation coefficient of the velocity fluctuations in the axial direction is calculated. According to the result

in the work of [1], the required pipe length to resolve the larges scale structures for the given Reynolds number should be $L = 2.5D = 125\ mm$, where L is the required length in parallel wall direction.

First, the approach is directed towards LES of channel flow in this thesis to get an initial insight for the pipe flow. The required number of grid points in turbulent channel flow can be estimated for DNS with the following equation:

$$N_L = \frac{L}{\Delta x} \sim \frac{10l}{\eta} = 10Re_l^{3/4}, \tag{2.49}$$

where Δx is the grid spacing, length scales L represents the flow geometry (e.g. the pipe diameter), η is the Kolmogorov length scale (smallest eddies), l $[m]$ represents the largest eddies in the flow, and $Re_l \approx Re_L/170 \approx 3421/170 \approx 20$ for water and $Re_l \approx Re_G/170 \approx 3632/170 \approx 21$ for air [3]. Thus, the total number of grid points for water in three dimensions N_L^3 is proportional to

$$N_L^3 \sim 10Re_l^{9/4} \sim 10(20)^{9/4} \sim 8.5 \times 10^3, \tag{2.50}$$

and for air

$$N_L^3 \sim 10Re_l^{9/4} \sim 10(21)^{9/4} \sim 9.5 \times 10^3. \tag{2.51}$$

According to the studies, at $Re_L = 3421$ the LES can be resolved near-wall region since the CPU time does not differ significantly compared to the modeled near-wall region case [9]. The number of grid points is proportional to

$$N_L = \frac{L}{\Delta x} = \frac{L}{l}\frac{l}{\Delta}\frac{\Delta}{\Delta x} \approx 20\frac{l}{\Delta}, \tag{2.52}$$

where the ratio of the flow geometry to the largest length scale is approximated as $L/l \approx 10$ for wall-bounded shear-driven turbulent flows [1]. The filter length Δ is assumed to be double the grid spacing, i.e. $\Delta \approx 2\Delta x$, in order to keep the range of grid scale motions as large as possible.

For LES, the ratio of l/Δ plays an important role in computations (instead of Re_l as it is the case for DNS). The computational cost increase rapidly as the ratio gets larger. Furthermore, LES computations give realistic results when the ratio is large (e.g. $l/\Delta > 10$). The reason behind this is the increase in the range of turbulent length scales that are resolved and the decrease in the range of length scales that are used in SGS stress tensor when the ratio increases. However, in the study of [1], the computations of LES are considered to be realistic when the ratio is even smaller (i.e. $l/\Delta > 2$). As the ratio gets smaller, the SGS closure model becomes important, whereas the range of resolved grid scale decreases. In the study of [1], when the value of the ratio is around 1, computations are considered to be unrealistic. The smallest value that gives realistic LES results used in study [1] is 1.8.

In this work, the ratio is taken as $l/\Delta = 5$, and the length scale in the radial direction for the pipe flow case L_r is equal to the pipe diameter $L_r = D = 50\ mm$ and in the

[3] In general case, the Reynolds number is expressed as $Re = U_b D/\nu$ where U_b being the mean velocity, D is the pipe diameter. The Reynolds number $Re_l = ul/\nu$ can be approximated by using ($l \sim \frac{1}{10}D$) and ($u \sim \frac{1}{17}U_b$), hence, $Re_l = ul/\nu \approx Re/170$.

streamwise direction $L_z = 2.5D = 125\ mm$. Hence, for the axial direction $l \approx 12.5\ mm$, and $\Delta = 2.5\ mm$, which results in the following number of grid points:

$$N_L \approx 20\frac{l}{\Delta} \approx 100. \tag{2.53}$$

The estimation of the number of grid points for DNS is explained in Appendix A and

	LES		**DNS**
Water	Uniform grid spacing	Non-uniform grid spacing	
N_x	2	16	56
N_y	7	32	75
N_z	7	24	75
N_{total}	98	12288	3.2×10^5
Air	Uniform grid spacing	Non-uniform grid spacing	
N_x	2	17	59
N_y	8	34	79
N_z	8	26	79
N_{total}	128	15028	3.7×10^5

Table 2.2: Comparison of LES and DNS for air flow with $Re_G = 3632$ and for water flow with $Re_G = 3421$

the estimation of the number of grid points for LES is explained in Appendix B in detail for the pipe flow under consideration. The number of grid points required for DNS of single phase water flow with non-uniform grid spacing in radial direction is $N_r = 11$, and with uniform grid spacing for spanwise and streamwise directions $N_\theta = 248$, $N_z = 198$ respectively. The total number of grid cells required for DNS for the liquid phase are approximately 5×10^5 (see Fig. 2.2). For the air case, the number of grid points are quite similar because of the small difference in the Reynolds numbers of liquid and gas phases.

For a well-resolved LES (i.e., resolved viscous sublayer), the near-wall grid resolution should be fine enough. The first grid-point should locate in the viscous sublayer for LES-NWR, i.e. $y^+ = 1$. The computation of the first LES case is done with grid points that are equally spaced, and the viscous sublayer is not resolved, i.e., the first grid-point is within the inertial sublayer ($y^+ = 32.8 > 30$). This approach does not need any additional damping since the first grid-point is far away from the pipe flow. The total number of grid points is quite small for water and air phases. The reason for this is that the boundary layer is almost fully modeled because of the large value of wall unit, $y^+ = 32.8$.

For the second case (the non-uniform case for $y^+ = 1.5$), the grid spacing is non-uniform only in the radial direction (normal to the wall) and three grid points are located within the viscous sublayer. The grid refinement factor for this configuration is calculated and validated in Appendix B.

2.3.2 Estimated Reynolds number for the available computational power

The problem can be simulated both with DNS and LES for the given Reynolds numbers with serial algorithms in a reasonable amount of time with the computational resources available in the Scientific Computing group of the Delft Institute for Applied Mathematics, $30 \times 150 \times 90$ grid points. However, the estimations show that LES is quite less demanding than DNS since the Reynolds number is relatively smaller for doing LES. Therefore, LES can be considered to be feasible for both water and air flows considering the computational complexity especially when the computational domain length is larger.

The maximum computational capacity is slightly exceeded with DNS. On the other hand, the maximum possible Reynolds number for LES can be approximated with the available computational power (i.e., $4{\times}10^5$ is the maximum number for total grid points).

For LES with non-uniform grid spacing, the Reynolds number used for the calculations is not large enough to exceed the computational limitations. When the Reynolds number is approximately equal to 4×10^4, then the required number of grid points in r-direction is $N_r \approx 33$. The dimensionless mesh width in r-direction for the calculated number of grid points is approximately equal to 0.0166. The approach implemented in the Appendix B is applied here and the required mesh width in θ- and z- directions can be calculated as 0.26166 and 0.02768 respectively. For these uniform grid spacings, the required number of grid points are 120 and 90 respectively for θ- and z-direction. The available computational power is almost fully used with these number of grid points that are obtained for the chosen Reynolds number value of 4×10^4.

Chapter 3

The physics of stratified two-phase flow

In this chapter, the physics of stratified two-phase flow is investigated. It is known that the turbulence due to shear in the interface behaves similarly as the turbulence in single-phase flows. Other than this, turbulence in two-phase flows may be quite different from single phase flows. In turbulent single phase flows, there is a need to model the Reynolds stress terms to obtain the closure relations. However, in two-phase flow many more closure relations are needed. Unlike single phase flow, there is also no universal model that describes the instability that causes the transition to another flow pattern in two-phase flows.

In the presence of waves, the turbulence behaves differently compared to the turbulence in shear layers. At the interface, the fluctuations occur mostly because of the effects of the wave motion. In two-phase flows, there are two main reasons for the formation of fluctuations. One of them is the turbulence created by shear stresses in near-wall regions, and the other one is induced by wave motions. The governing equations for the two-phase stratified flow are discussed in this chapter. The equations for the two-phase flow model are assumed to be identical to the single-phase turbulence model under the assumption that the interface has little influence.

3.1 Dimensionless parameters

In order to identify the dominant forces acting on the flow, these forces need to be compared using dimensionless groups. This way some forces may be neglected and the problem becomes easier to handle. Dimensionless variables are defined by a characteristic length L, a velocity scale U, and a time scale τ;

$$x = Lx^*, \quad u = Uu^*, \quad t = \tau t^*. \tag{3.1}$$

where x^*, t^*, and u^* are the dimensionless variables. The *Reynolds number Re*, which describes the ratio of inertial forces to the viscous forces, is defined by:

$$Re = \frac{\rho LU}{\mu} = \frac{LU}{\nu}. \tag{3.2}$$

where ρ is the fluid density, L is the characteristic domain length, U is the fluid velocity and μ is the dynamic viscosity. The Reynolds number represents the flow pattern (i.e. laminar or turbulent) and is certainly an important parameter for this project. The *Capillary number Ca* represents the ratio of viscous forces to surface tension forces and is equal to:

$$Ca = \frac{\mu U}{\gamma}. \tag{3.3}$$

where γ is the surface tension coefficient. The *Morton number Mo*, which denotes the shape of bubbles or drops in a fluid, for the case of a bubble with a negligible inner density can be defined as:

$$Mo = \frac{g\mu^4}{\rho\gamma^3}. \tag{3.4}$$

where g is the acceleration of gravity. Both of these numbers can be used to verify that the effects of surface tension are negligible.

3.2 Governing equations and boundary conditions at the interface

The finite thickness layer, where the material properties change, between two phases is called the *interface*. Surface tension occurs at the interface due to the attractive forces acting on molecules. In two-phase flows, the change in viscosity due to different material properties and the existence of surface tension forces at the interface lead to jumps and discontinuities at the interface for pressure and the gradient of the velocity field.

In order to observe the influence of the interface, phases 1 and 2 are compared at the boundary S, which separates them. When there occurs a phase change at the interface, there will be a mass flux $\dot{m}$ through the boundary. Conservation of mass gives

$$\dot{m} \equiv \rho_1 \left(\boldsymbol{u_1} - \boldsymbol{w}\right) \cdot \boldsymbol{n} = \rho_2 \left(\boldsymbol{u_2} - \boldsymbol{w}\right) \cdot \boldsymbol{n}, \tag{3.5}$$

where $\mathbf{n}$ is the unit normal and $\boldsymbol{w}.\mathbf{n}$ is the normal velocity of the interface. At first, if the interface is defined as

$$S(x,t) = 0, \tag{3.6}$$

after time dt, the surface can be expressed by taking derivative of Eq. (3.6):

$$\frac{\partial S}{\partial t} + \boldsymbol{w} \cdot \nabla S = 0. \tag{3.7}$$

By defining the unit normal as

$$\boldsymbol{n} = \frac{\nabla S}{|\nabla S|}, \tag{3.8}$$

the expression for $\boldsymbol{w} \cdot \boldsymbol{n}$ is obtained:

$$\boldsymbol{n} \cdot \boldsymbol{w} = -\frac{1}{|\nabla S|} \frac{\partial S}{\partial t}. \tag{3.9}$$

The physical parameters vary through the interface. The thickness of the interface that separates the bulk parts of two fluids is much smaller than other length scales. Thus, it is reasonable to use a functional interface, in which there are two boundary conditions for the interface; the *kinematic boundary condition* and the *dynamic boundary condition* [10]. If there is no mass transfer, then at $S = 0$ the surface is impermeable $\dot{m} = 0$, thus, $\boldsymbol{n} \cdot \boldsymbol{u} = \boldsymbol{n} \cdot \boldsymbol{w}$. The equation for the kinematic boundary condition Eq. (3.9) is modified as:

$$\frac{\partial S}{\partial t} + \boldsymbol{u} \cdot \nabla S = 0, \tag{3.10}$$

and it is called the *kinematic boundary condition*. The tangential velocities at the interface for both fluids should be in balance:

$$\boldsymbol{n} \times (\boldsymbol{u} - \boldsymbol{w}) = 0. \tag{3.11}$$

By using eqs. (3.7) and (3.11), it can be shown that:

$$\boldsymbol{u} = \boldsymbol{w}. \tag{3.12}$$

For steady state flow, the kinematic boundary condition can be expressed as:

$$\boldsymbol{u} \cdot \boldsymbol{n} = 0. \tag{3.13}$$

Another condition for the interface is that the momentum (the pressure and viscous stresses) is balanced through the interface by the force acting due to surface tension. When two fluids are viscous, the tangential velocity across the interface is continuous (when $\dot{m} = 0$):

$$\boldsymbol{u_1} = \boldsymbol{u_2}. \tag{3.14}$$

This is called the *dynamic boundary condition*:

$$\boldsymbol{n} \cdot T_1 - \boldsymbol{n} \cdot T_2 = \sigma \boldsymbol{n} \left(\nabla_s \cdot \boldsymbol{n}\right) - \nabla_s \sigma, \tag{3.15}$$

where $\boldsymbol{T} = \boldsymbol{\tau} - \boldsymbol{p}\mathbb{1} = \mu \left(\nabla \boldsymbol{u} + \nabla \boldsymbol{u^T}\right) - \boldsymbol{p}\mathbb{1}$ is the stress tensor, $\mathbb{1}$ is the identity tensor, $\nabla_s = (\mathbb{1} - \boldsymbol{n_1 n_1}) \cdot \nabla$ is the surface gradient operator and σ is the surface tension. When there is no shear stress, the expression for static interfaces is (Young-Laplace)

$$p_1 - p_2 = \sigma \mathcal{K} = \sigma \left(\frac{1}{R_1} + \frac{1}{R_2}\right), \tag{3.16}$$

where $\mathcal{K} = \nabla_s \cdot \boldsymbol{n} = \left(\frac{1}{R_1} + \frac{1}{R_2}\right)$ is the mean curvature of the interface. Normal stress balance is

$$\boldsymbol{n} \cdot T_1 \cdot \boldsymbol{n} - \boldsymbol{n} \cdot T_2 \cdot \boldsymbol{n} = \sigma \left(\nabla_s \cdot \boldsymbol{n}\right). \tag{3.17}$$

The tangential stress balance with tangential unit vector $\boldsymbol{t}$ is

$$\boldsymbol{n} \cdot T_1 \cdot \boldsymbol{t} - \boldsymbol{n} \cdot T_2 \cdot \boldsymbol{t} = -\nabla \sigma \cdot \boldsymbol{t}. \tag{3.18}$$

3.3 Overview of the governing equations

The flow is incompressible, both individual phases are Newtonian and immiscible. The phase with the higher density flows on the bottom of the pipe. The governing equations for the two-phase stratified flow are the Navier-Stokes equations in both domains Ω_α and Ω_γ (see Fig. 3.1). The equations for the two-phase flow model are nearly identical to the equations for the single phase case. The only differences are the variable density and viscosity. Therefore, it is reasonable to apply a single-phase turbulence model under the assumption that the interface has little influence. The continuity equation for a phase with density ρ_α is as follows:

$$\nabla \cdot \boldsymbol{u_\alpha} = 0. \tag{3.19}$$

The conservation equation for momentum is given by

$$\frac{\partial u_{\alpha,i}}{\partial t} + u_{\alpha,j}\frac{\partial u_{\alpha,i}}{\partial x_j} = -\frac{1}{\rho}\frac{\partial}{\partial x_i}(\overline{p}_\alpha + \frac{2}{3}\rho k) + \frac{\partial}{\partial x_j}\left[\nu_{eff}\left(\frac{\partial u_{\alpha,i}}{\partial x_j} + \frac{\partial u_{\alpha,j}}{\partial x_i}\right)\right], \tag{3.20}$$

where ν_{eff} is the effective kinematic viscosity, $u_{\alpha,i}$ and $\overline{p}_\alpha$ represent the velocity and pressure of the phase α respectively. Eqs. (3.19) and (3.20) for phase α are subject to the

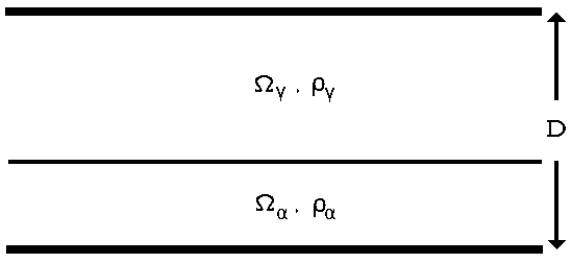


Figure 3.1: Schematic representation of stratified pipe flow with a pipe diameter D and phases γ and α

following boundary conditions (at the interface S between phase α and γ, see Fig. 3.2):

$$\rho_\alpha(u_{\alpha,i} - w)\cdot\boldsymbol{n}_\alpha + \rho_\gamma(u_{\gamma,i} - w)\cdot\boldsymbol{n}_\gamma = 0, \tag{3.21}$$

$$\rho_\alpha u_{\alpha,i}(u_{\alpha,i} - w)\cdot\boldsymbol{n}_\alpha + \rho_\gamma u_{\gamma,i}(u_{\gamma,i} - w)\cdot\boldsymbol{n}_\gamma = \tag{3.22}$$

$$(-p_\alpha\mathbb{1} + \tau_\alpha)\cdot\boldsymbol{n}_\alpha + (-p_\gamma\mathbb{1} + \tau_\gamma)\cdot\boldsymbol{n}_\gamma - \nabla_S\sigma_{\alpha\gamma} + \frac{2\sigma_{\alpha\gamma}}{|\nabla S|}\nabla S.$$

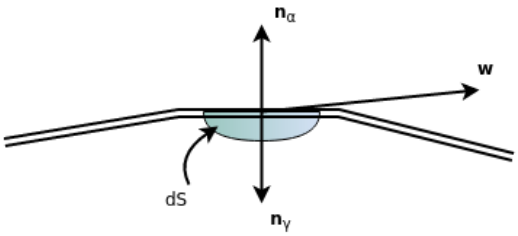


Figure 3.2: Representation of a fraction of the interface between phases α and γ

3.4 General Overview

The model for two-phase flow leads to the variable density and viscosity Navier-Stokes equations in the absence of surface tension. The value of the *Capillary number Ca* and the *Morton number Mo* indicate that surface tension effects can be neglected due to the fact that viscous forces dominate surface tension forces for high-Re flows. As a result, the equations for two-phase flow become identical to the Navier-Stokes equations with variable density and viscosity for single-phase flow. In addition, it can be shown that the model for two-phase flow is identical to Eqs. (3.19) and (3.20).

Chapter 4

Turbulence modeling for turbulent two-phase flows

In this chapter, the computational techniques used in the literature for turbulent two-phase flows are investigated. The computational effort required for DNS is not significantly more than LES for low Reynolds number (i.e. $Re \approx 3000$) flows. It was realized during the estimation procedure that modeling the problem with DNS is realistic even for the two-phase case. However, unlike LES, DNS is not feasible for higher Reynolds number flows. Therefore, LES of turbulent two-phase flow is the main focus of this thesis in order to predict the onset of instability of the interface and the formation of slug.

Most of the DNS and LES studies in the literature are carried out for channel flows. Nevertheless, there are also few studies using LES for pipe flows. However, simulation of turbulent two-phase stratified pipe flows is even more sparse, and LES is not a widely used turbulence model for that case.

4.1 Literature review

The earliest study of LES for single phase pipe flow is the work of Eggels et al. [11]. Eggels also performed LES computations for single phase pipe flows [1], and the results of DNS and LES were compared with experimental results. Eggels et al.[12] used LES for rotating turbulent single phase pipe flow, and LES of turbulent curved single phase pipe flow was carried out by [13]. LES study for turbulent two-phase stratified pipe flow is very limited. Therefore, initially in this thesis, LES studies that are not necessarily for pipe flow and do not include momentum transfer between two different phases are considered. In addition to this, studies about other turbulence models are also compared since LES is not the common method for modeling turbulent two-phase stratified pipe flow.

In the master thesis of [14], the turbulence behavior of the interface in stratified two-phase flow has modeled with the classical $k-\omega$ low Reynolds number correction model and with

the modified $k - \omega$ model in order to predict more accurately. Available works in the literature about stratified two-phase flow with RANS have been analyzed and compared in this thesis. In this thesis, the flow has been simulated in 1-D and a new input parameter, which represents the wavy motion at the interface, has implemented to the model to obtain results close to the experimental results. In order to realize the momentum transfer in between two phases, necessary adjustments were made to the grid and to the model. The turbulence near the gas side of the interface is assumed to be similar to the one near the wall, which was indicated by the DNS studies of [15] and [16]. The thesis has pointed out that there is no optimum and robust method that simulates turbulent two-phase stratified flow.

In the study of [17], the flow in both phases and turbulence at the interface is resolved using the complete filtered two-phase flow governing equations. The mathematical formulation was based on one-fluid model, which means that the interaction between two phases needs to be modeled. The filtering and averaging operations yield terms for turbulence and interface. According to the study, LES is reasonably accurate when the small turbulent scales are much smaller than the scale of the interface and it is very hard to analyze the behavior of the turbulence at the interface. A front-tracking method with a defined sharp interface, which does not use the usual smoothing function of the interface, was used in the study to model the interface explicitly. Results showed that the volume filtering method overestimates the momentum transfer, mass-weighted filtering process underestimates the momentum transfer. Therefore, the work recommends the use of a mass-weighted filtering process for modeling two-phase turbulent flow. The interfacial terms are very important for the turbulence behavior of the phases, and the interface can absorb some of the energy of the turbulence and advects it to the other phase. The study has also shown that the inertia term cannot be modeled with a viscosity assumption and it was exhibited different behavior than the single-phase flow near the interface.

The turbulent free-surface flow in an unbaffled stirred tank reactor is modeled with LES in the work of [18]. The study emphasizes the limitations of RANS methods compared to Reynolds stress models and LES (in terms of the mean flow characteristics) for the given problem, which consists of rotation of the stirrer. The stirrer was modeled by using an immersed boundary method and a front-tracking method was used to capture the free-surface vortex. In the work, most of the mean and fluctuation characteristics were successfully computed by LES.

In the study of [19], a modified two-layer turbulence model is used in order to account for the wavy interface. The turbulence model consists of a two-equation k-ϵ model and a one-equation k-l model with an interfacial roughness has provided the results for the interfacial shear stress without the use of wall functions. In the method, the immersed interface method has been used to treat interfacial boundary conditions. A parameter, which has a small effect on the results, has been used to represent the interfacial roughness. The results of the study agree well with experiments.

Ghorai et al. [4] stated in their study that modeling of smooth stratified flow is not a problem since the smooth friction factor can be easily expressed. However, for wavy stratified flow, there are two possible approaches to take: First one is to propose a global empirical correlation for the interfacial friction factor, and the second approach is to define the interface as an interfacial roughness by using the idea behind shear flow. The interfacial roughness has been estimated by defining a ratio between the interfacial friction

factor and the wall friction factor.

In the work of [20], the momentum transfer at the interface was considered to be the crucial issue in modeling of two-phase (gas-liquid) stratified flows. The main approach used in most of the studies and also in this one was to extrapolate the information gathered from single phase flow and to obtain empirical correlations based on experimental data. According to the study, the values of the liquid wall shear and the interfacial shear was increased as waves get larger compared to the values predicted for smooth stratified flow. The results of the study shows that the interaction between the phases can be expressed with the closure relations that were used in the study.

The numerical simulation of slug flow in horizontal pipes was carried out by [21]. Two-fluid approach was used in the study and it was showed that the wall friction of the liquid phase has a significant affect on the formation of slugs. The formation of slugs also depends on the perturbation in the inlet boundary condition and the length of the computational domain.

Lakehal et al. [22] has examined the development of large-wave structures in two-phase turbulent pipe flow using the Level Set method for interface tracking and *very-large eddy simulation (VLES)* for turbulence modeling. The VLES used in the study is based on a $k - \epsilon$ subgrid model in order to include the effect of sub-scale turbulence. The reason for a more detailed subgrid scale turbulence model compared to a zero-equation model like in LES is to filter a larger part of turbulent fluctuations. In the study of [23], LES and VLES methods have been used for turbulence modeling and Level Set method has been used to model the free surface flow, for interface tracking. it has shown that the Level Set method predicts the transition of a gas-liquid stratified flow to slug flow with an accuracy that is better than the two-fluid phase-average model.

In general, modeling turbulence in stratified two-phase flow is challenging if there is a need to track the interface accurately and also if surface tension effects are not negligible. In this case, RANS models do not yield reliable outcomes and DNS is impractical for large Re flows. Therefore, VLES or LES methods are the promising methods in terms of computational complexity and accuracy.

Chapter 5

Computation of fully developed laminar pipe flow

In this chapter, boundary conditions and several test cases are discussed for validation purposes. The velocity of the flow is periodic but the pressure is not. Thus, the algorithm, which is not periodic for pressure, is modified such that the new algorithm is periodic not only in circumferential direction but also in axial direction for velocity and also for pressure. The discretization used in the algorithm is able to represent the solution of the Poiseuille profile exactly (except the wall boundary condition which is first order accurate) for the constant viscosity case.

5.1 Introduction

The physical problem consists of a channel (for 2D) or a pipe (for 3D) connected to two reservoirs. The pressure at the inlet is higher than the pressure at the outlet reservoir, which gives rise to a flow through the channel or pipe from inlet to outlet. The velocity profile is assumed to be fully developed at all locations along the channel or pipe and the pressure difference between left and right boundaries is constant. The solution for a given pressure drop or for a given flow rate is uniquely defined. Prescribing either of these quantities requires the other to be solved for simultaneously with the velocity profile.

For simplicity, the problem is first considered only in two dimensions with periodicity at the inlet and outlet boundaries of a rectangular domain as shown in Fig. 5.1. The boundary conditions for the velocity and pressure in a channel are given by

$$\boldsymbol{u}(x_1, x_2)\Big|_{inlet} = \boldsymbol{u}(x_1 + L, x_2)\Big|_{outlet}, \tag{5.1}$$

$$\frac{\partial \boldsymbol{u}(x_1, x_2)}{\partial n}|_{inlet} = -\frac{\partial \boldsymbol{u}(x_1 + L, x_2)}{\partial n}|_{outlet}, \tag{5.2}$$

$$p(x_1, x_2)_{inlet} = p(x_1 + L, x_2)_{outlet} + c. \tag{5.3}$$

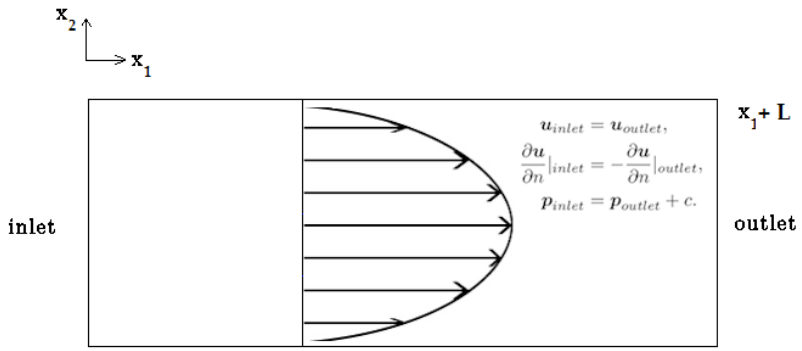


Figure 5.1: Schematic for a fully-developed laminar flow with periodicity at a rectangular domain

where x_1 and x_2 represent the horizontal and vertical directions respectively, L denotes the channel length, $\boldsymbol{u}$ is the Cartesian velocity vector (in $2D$), p is the pressure, $\boldsymbol{n}$ is the outward directed normal and c represents the unknown constant jump in pressure, which stands for the periodicity of the pressure.

The decrease in pressure along the channel or pipe is dependent on the velocity profile and is not known *a priori*. In order to obtain the unknown constant pressure difference c implicitly, the flow rate Q needs to be specified and it is defined by

$$Q = -\int_{A_{inlet}} \boldsymbol{u} \cdot \boldsymbol{n}\, dA. \tag{5.4}$$

5.2 Poiseuille channel flow

First, the problem is considered with an assumption that the computational domain is two-dimensional, in which the horizontal (x_1-direction) velocity only varies in vertical direction and the vertical (x_2-direction) velocity assumed to be zero. In order to implement the boundary conditions (i.e., the periodic and no-slip boundary conditions), necessary modifications are made within the code. The conservation of mass (continuity equation) and momentum equations in Cartesian coordinates for an incompressible flow are

$$\nabla \cdot \boldsymbol{u} = \frac{\partial u_i}{\partial x_i} = 0, \tag{5.5}$$

$$\rho \frac{\partial u_i}{\partial t} = \mu \frac{\partial^2 u_i}{\partial x_i x_j} - \frac{\partial p}{\partial x_i}. \tag{5.6}$$

where ρ and μ are assumed constant and $\boldsymbol{u} = (u_x, u_y)$. Equations 5.5 and 5.6 can be written as a differential-algebraic system of the following structure:

$$\boldsymbol{D}\boldsymbol{u} = 0, \tag{5.7}$$

$$\boldsymbol{M}\boldsymbol{u} = \boldsymbol{S}\boldsymbol{u} - \boldsymbol{G}p - \bar{\boldsymbol{G}}c. \tag{5.8}$$

Here $\boldsymbol{D}$ is a linear algebraic operator representing the discretization of divergence operator in the continuity equation, $\boldsymbol{M}\boldsymbol{u}$ gives the discretization of the time derivative, $\boldsymbol{S}$ is a linear

algebraic operator arising from the discretization of the viscous terms, and $\boldsymbol{G}$ is a linear algebraic operator representing the discretization of the pressure gradient except for the unknown constant, whose contribution is stored in $\bar{\boldsymbol{G}}$.

Discretization of Eq. (5.4) gives a linear relation:

$$\boldsymbol{R}\boldsymbol{u} = Q, \tag{5.9}$$

where $\boldsymbol{R}$ has non-zero contributions for all normal components of the velocity at the inlet.

The horizontal velocity vector component $(u_1)_{ij}$ with two indices can be mapped into the single index velocity vector notation $(u_1)_k$. For the normal velocity component the mapping is as follows:

$$(u_1)_{ij} = (u_1)_{(j-1)(nx)+i} \quad \text{for} \quad j = 1, ..., ny, \quad i = 1, ..., nx. \tag{5.10}$$

where nx and ny are the number of identical intervals to construct a mesh of equally spaced points [1] in horizontal (x_1) and vertical (x_2) directions respectively as shown in Fig. 5.2. The mapping for the velocity in the vertical direction is

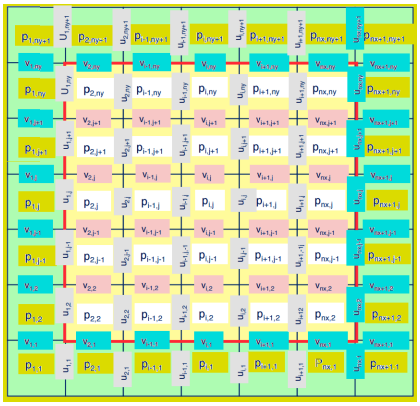


Figure 5.2: Staggered grid with $(nx-1) \times (ny-1)$ control volumes: The square with red borders represent the computational domain where velocity and pressure components are unknown except the vertical velocities on the upper and lower boundaries and the horizontal velocity components on the right boundary due to periodicity

$$(u_2)_{ij} = (u_2)_{(j-1)(nx)+(i-1)} \quad \text{for} \quad j = 1, ..., ny-1, \quad i = 2, ..., nx+1. \tag{5.11}$$

Although, it is known *a priori* that u_2 is zero, it is still going to be solved in order to make the discretization similar to the 3D algorithm for turbulent flow where u_2 will have non-zero fluctuations. The length of the velocity vector $\boldsymbol{u}$ is $[(nx)ny + nx(ny-1)] \times 1$ the numbering suggests there are actually $N \times 1$ (where $N = [ny(nx) + nx(ny-1)]$). The vector $\boldsymbol{R}$ has non-zero entries at every $((j-1)(nx)+1)$th element of velocity vector for $j = 1, ..., ny$. The size of $\boldsymbol{R}$ is $1 \times N$.

$\boldsymbol{M}$ is a diagonal matrix which consists of values of ρ multiplied by the area of the corresponding control volume [2]. The size of $\boldsymbol{M}$ is $N \times N$. First $ny(nx)$ rows are for the

[1] $x_i = ih_1$, $i = 0, ..., nx$, where h_1 is the horizontal mesh width given by $h_1 = L/nx$ and L is the horizontal length of the domain.

[2] Area of the control volume is multiplication of mesh width in horizontal and vertical direction, $A = h_1 h_2$.

horizontal velocity and last $nx(ny-1)$ rows are for the vertical velocity. There are ny entries of matrix $\boldsymbol{A}$ as diagonal elements in first $ny(nx)$ rows of the matrix $\boldsymbol{M}$:

$$\boldsymbol{A}_{(\boldsymbol{ny})\times(\boldsymbol{nx})} = \begin{bmatrix} 1 & 0 & \dots & \dots & 0 \\ 0 & 1 & 0 & \dots & 0 \\ \vdots & 0 & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & 1 & 0 \\ 0 & \dots & \dots & 0 & 1 \end{bmatrix}.$$

There are $(ny-1)$ entries of matrix $\boldsymbol{B}$ as diagonal elements in last $nx(ny-1)$ rows of the matrix $\boldsymbol{M}$:

$$\boldsymbol{B}_{(\boldsymbol{nx})\times(\boldsymbol{ny-1})} = \begin{bmatrix} 1 & 0 & \dots & \dots & 0 \\ 0 & 1 & 0 & \dots & 0 \\ \vdots & 0 & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & 1 & 0 \\ 0 & \dots & \dots & 0 & 1 \end{bmatrix}.$$

Accordingly, $\boldsymbol{M}$ looks as follows:

$$\boldsymbol{M} = \rho h_1 h_2 \begin{bmatrix} \overbrace{A & 0 & \dots & \dots & 0}^{\text{ny(nx)}} & \overbrace{0 & \dots & \dots & \dots & 0}^{\text{nx(ny-1)}} \\ 0 & A & 0 & \dots & 0 & 0 & 0 & \dots & \dots & 0 \\ \vdots & 0 & \ddots & \ddots & \vdots & \vdots & \vdots & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & A & 0 & \vdots & \vdots & \ddots & 0 & 0 \\ 0 & \dots & \dots & 0 & A & 0 & \dots & \dots & 0 & 0 \\ 0 & \dots & \dots & \dots & 0 & B & 0 & \dots & \dots & 0 \\ \vdots & \ddots & \ddots & \dots & 0 & 0 & B & 0 & \dots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots & \vdots & 0 & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & 0 & 0 & \vdots & \vdots & \ddots & B & 0 \\ 0 & \dots & \dots & 0 & 0 & 0 & \dots & \dots & 0 & B \end{bmatrix} \begin{matrix} r{=}1 \\ r{=}2 \\ \dots \\ \dots \\ r{=}(nx)ny \\ r{=}(nx)ny \\ +1 \\ r{=}(nx)ny \\ +2 \\ \dots \\ \dots \\ r{=}(nx)ny \\ +nx(ny{-}1) \end{matrix}.$$

The coefficients of matrix $\boldsymbol{M}$ have half the normal magnitude at every $((j-1)(nx)+k)$th row and column index because the control volumes at left and right boundary cells are half the inner control volumes, where $k = 1, nx$ for every $j = 1, ..., ny$. Due to the periodic boundary condition, Eq. (5.45), the coefficient of u_{outlet} is added to the coefficient of u_{inlet} and coefficients of u_{inlet} become 1 instead of 1/2.

$\boldsymbol{Du} = 0$ represents the discretization of the continuity equation and each row of it corresponds to the discretization of the continuity equation for one control volume. The first $ny(nx)$ columns of $\boldsymbol{D}$ matrix represent the coefficients for the x_1-direction (horizontal) velocity components and the last $(ny-1)nx$ columns represent the coefficients for the x_2-direction (vertical) velocity components:

$$\boldsymbol{D}_{[(\boldsymbol{nx})(\boldsymbol{ny})]\times\boldsymbol{N}} = \begin{bmatrix} \boldsymbol{D^1} & \boldsymbol{D^2} \\ \boldsymbol{D^3} & \boldsymbol{D^4} \\ \boldsymbol{D^5} & \boldsymbol{D^6} \end{bmatrix}.$$

$\boldsymbol{D^1}$ and $\boldsymbol{D^2}$ represent the discretization of the bottom control volumes for the x_1 and x_2 direction velocity vectors:

$$\boldsymbol{D^1_{(nx)\times[(nx)ny]}} = h_2 \begin{bmatrix} \overbrace{-1 & 1 & 0 & \dots & \dots}^{\text{(nx)}} & \overbrace{\dots & \dots & 0}^{\text{(nx)(ny-1)}} \\ 0 & -1 & 1 & 0 & \dots & \dots & \dots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \vdots & \vdots & \vdots \\ 1 & 0 & \dots & -1 & 0 & 0 & \dots & 0 \end{bmatrix},$$

$$\boldsymbol{D^2_{(nx)\times[nx(ny-1)]}} = h_1 \begin{bmatrix} \overbrace{1 & 0 & \dots & \dots}^{\text{nx}} & \overbrace{\dots & \dots & 0}^{\text{nx(ny-2)}} \\ 0 & 1 & 0 & \dots & \dots & \dots & 0 \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots & \vdots \\ 0 & \dots & \dots & 1 & 0 & \dots & 0 \end{bmatrix}.$$

$\boldsymbol{D^3}$ and $\boldsymbol{D^4}$ consist of the coefficients for x_1 and x_2 direction velocity vectors in the inner control volumes:

$$\boldsymbol{D^3_{[nx(ny-2)\times nxny]}} = h_2 \begin{bmatrix} \overbrace{0 & \dots & 0}^{\text{nx}} & \overbrace{-1 & 1 & 0 & \dots & \dots}^{\text{(nx)(ny-2)}} & \overbrace{0 & \dots & 0}^{\text{nx}} \\ \vdots & \dots & \vdots & 0 & -1 & 1 & 0 & \dots & \vdots & \dots & \vdots \\ \vdots & \dots & \vdots & \vdots & \ddots & \ddots & \ddots & \ddots & \vdots & \dots & \vdots \\ 0 & \dots & \dots & 0 & 1 & \dots & -1 & 0 & 0 & \dots & 0 \end{bmatrix},$$

$$\boldsymbol{D^4_{nx(ny-2)\times[nx(ny-1)]}} = h_1 \begin{bmatrix} \overbrace{-1 & 0 & \dots}^{\text{nx}} & \overbrace{1 & 0 & \dots & \dots & 0}^{\text{nx(ny-2)}} \\ 0 & -1 & 0 & \dots & 1 & 0 & \dots & \vdots \\ \vdots & \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & \dots & \dots & -1 & 0 & \dots & 1 & 0 \\ 0 & \dots & \dots & \dots & -1 & 0 & \dots & 1 \end{bmatrix}.$$

$\boldsymbol{D^5}$ and $\boldsymbol{D^6}$ represent the top boundary control volumes and are defined by

$$\boldsymbol{D^5_{(nx)\times[(nx)ny]}} = h_2 \begin{bmatrix} \overbrace{0 & \dots & \dots & 0}^{\text{(nx)(ny-1)}} & \overbrace{-1 & 1 & 0 & \dots & 0}^{\text{nx}} \\ \vdots & \dots & \dots & \vdots & 0 & -1 & 1 & 0 & \vdots \\ \vdots & \dots & \dots & \vdots & \vdots & \vdots & \ddots & \ddots & \vdots \\ \vdots & \dots & \dots & \vdots & 0 & \dots & \dots & -1 & 1 \\ 0 & \dots & \dots & 0 & 1 & 0 & \dots & 0 & -1 \end{bmatrix},$$

$$\boldsymbol{D}^{\boldsymbol{6}}_{\boldsymbol{(nx)\times[nx(ny-1)]}} = h_1 \begin{bmatrix} \overbrace{\begin{matrix} 0 & \dots & \dots & 0 \\ \vdots & \dots & \dots & \vdots \\ \vdots & \dots & \dots & \vdots \\ \vdots & \dots & \dots & \vdots \\ 0 & \dots & \dots & 0 \end{matrix}}^{\text{nx(ny-2)}} & \overbrace{\begin{matrix} -1 & 0 & \dots & \dots & 0 \\ 0 & -1 & 0 & \dots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & 0 & -1 & 0 \\ 0 & \dots & \dots & 0 & -1 \end{matrix}}^{\text{nx}} \end{bmatrix}.$$

The matrix $\boldsymbol{S}$, which represents the discretization of the viscous terms, has zero entries for the vertical velocity component since the velocity in the vertical direction (the x_2 direction) is assumed not to be changing (i.e. it is constant and equal to 0). $\boldsymbol{S}$ can be represented with three sub matrices:

$$\boldsymbol{S}_{\boldsymbol{N\times N}} = \frac{1}{h_2^2} \begin{bmatrix} \boldsymbol{S^1} & \boldsymbol{S^4} \\ \boldsymbol{S^2} & \boldsymbol{S^5} \\ \boldsymbol{S^3} & \boldsymbol{S^4} \\ \boldsymbol{S^6} & \end{bmatrix}.$$

First, before defining the coefficient matrix for discretization of the viscous terms, the ghost velocity nodes need to be expressed in terms of known velocity values using Lagrange polynomial. There is no error in the numerical solution in the inner region of the domain and with a constant viscosity because the local truncation error is zero everywhere for this case. Therefore, it becomes important to handle the right hand side in an exact way as well. Linear interpolation will yield an estimation for the ghost value with an error since the velocity profile is parabolic. On the other hand, Lagrange polynomial will give an exact result for the specified parabolic velocity profile. In the Fig. 5.3, u_0 represents

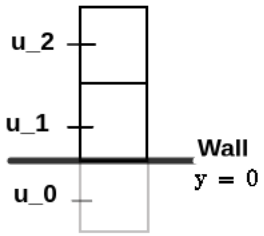


Figure 5.3: Ghost nodes outside the boundary

the ghost point outside the boundary, which is unknown. It can be expressed in terms of known velocity points u_1 and u_2 as follows:

$$u(y) = f(u_0, u_1, u_2), \tag{5.12}$$

$$u(y) = u_0 \frac{(y-\frac{h_2}{2})(y-\frac{3h_2}{2})}{(-\frac{h_2}{2}-\frac{hy}{2})(-\frac{h_2}{2}-\frac{3h_2}{2})} + u_1 \frac{(y-\frac{3h_2}{2})(y+\frac{h_2}{2})}{(\frac{h_2}{2}+\frac{h_2}{2})(\frac{h_2}{2}-\frac{3h_2}{2})} + u_2 \frac{(y-\frac{h_2}{2})(y+\frac{h_2}{2})}{(\frac{3h_2}{2}-\frac{h_2}{2})(\frac{3h_2}{2}+\frac{h_2}{2})},$$

where $-\frac{h_2}{2}$, $\frac{h_2}{2}$ and $\frac{3h_2}{2}$ are distances of u_0, u_1 and u_2 velocity points from the wall respectively and $\frac{h_2}{2}$ being half the mesh width in direction x_2. At the wall, (i.e. $y = 0$)

the *no-slip* boundary condition is applied, $u|_{y=0} = 0$:

$$u(0) = u_0 \frac{(0-\frac{h_2}{2})(0-\frac{3h_2}{2})}{(-\frac{h_2}{2}-\frac{h_2}{2})(-\frac{h_2}{2}-\frac{3h_2}{2})} + u_1 \frac{(0-\frac{3h_2}{2})(0+\frac{h_2}{2})}{(\frac{h_2}{2}+\frac{h_2}{2})(\frac{h_2}{2}-\frac{3h_2}{2})} + u_2 \frac{(0-\frac{h_2}{2})(0+\frac{h_2}{2})}{(\frac{3h_2}{2}-\frac{h_2}{2})(\frac{3h_2}{2}+\frac{h_2}{2})} = 0. \tag{5.13}$$

Eq. (5.13) yields the following expression for the ghost values, which holds at every i th row:

$$(u_1)_{i,0} = -2(u_1)_{i,1} + \frac{(u_1)_{i,2}}{3}. \tag{5.14}$$

In this case, the velocity component in x_1 direction $(u_1)_{i,2} = (u_1)_{(nx)+i}$ follows from the mapping defined earlier. Nonetheless, linear interpolation is used in order to obtain stable results for estimating ghost points:

$$(u_1)_{i,0} = -(u_1)_{i,1}. \tag{5.15}$$

$\boldsymbol{S^1_{nx\times[(nx)ny]}}$ represents the first (nx)th rows of $\boldsymbol{S}$ corresponding to the coefficients for the velocity in the x_1 direction, where $\alpha = -3 - 2\frac{h_2^2}{h_1^2}$ and $\beta = \frac{h_2^2}{h_1^2}$, though, the horizontal velocity does not change in axial direction it is still included in the matrix:

$$\boldsymbol{S^1} = \begin{bmatrix} \overbrace{\alpha & \beta & 0 & \dots & \beta}^{\text{nx}} & \overbrace{1 & 0 & \dots & \dots & \dots & 0}^{\text{nx}} & \overbrace{0 & \dots & \dots & 0}^{\text{nx(ny-2)}} \\ \beta & \alpha & \beta & 0 & \dots & 0 & 1 & 0 & \dots & \dots & \vdots & \vdots & \dots & \dots & \vdots \\ 0 & \beta & \alpha & \beta & 0 & \vdots & 0 & 1 & 0 & \dots & \vdots & \vdots & \dots & \dots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots & \vdots & \ddots & \ddots & \ddots & \ddots & \vdots & \vdots & \dots & \dots & \vdots \\ \beta & \dots & 0 & \beta & \alpha & 0 & \dots & \dots & 0 & 1 & 0 & \vdots & \dots & \dots & \vdots \end{bmatrix}.$$

$\boldsymbol{S^2_{[nx(ny-2)]\times[(nx)ny]}}$ represents the inner $nx(ny-2)$th rows of $\boldsymbol{S}$ which corresponds to the coefficients for the horizontal velocity. The matrix $\boldsymbol{S^2_{[(nx)]\times[3(nx)]}}$ with $\gamma = -2 - 2\frac{h_2^2}{h_1^2}$ is given by

$$\boldsymbol{S^2} = \begin{bmatrix} \overbrace{1 & 0 & \dots & \dots}^{\text{(nx)}} & \overbrace{\gamma & \beta & 0 & \dots & \beta}^{\text{ny(ny-2)}} & \overbrace{1 & 0 & \dots & \dots}^{\text{(nx)}} & 0 \\ 0 & 1 & 0 & \dots & \beta & \gamma & \beta & 0 & \dots & 0 & 1 & 0 & \dots & \vdots \\ \vdots & 0 & 1 & 0 & \dots & \beta & \gamma & \beta & 0 & \vdots & 0 & 1 & \dots & \vdots \\ \vdots & \dots & 0 & 1 & \beta & 0 & \dots & \gamma & \beta & \vdots & \vdots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \dots & \ddots & \ddots & \vdots \\ 0 & \dots & \dots & \dots & 0 & 1 & \beta & \dots & \beta & \gamma & 0 & \dots & 1 & 0 \end{bmatrix}.$$

$\boldsymbol{S^3_{nx\times[(nx)ny]}}$ shows the last $(nx-1)$th rows of $\boldsymbol{S}$ corresponding to the coefficients for the horizontal velocity:

$$\boldsymbol{S^3} = \begin{bmatrix} \overbrace{0 & \dots & 0 & 1 & 0 & \dots & \dots & \dots & \dots}^{\text{nx(ny-2)}} & \overbrace{\alpha & \beta & 0 & \dots & \beta}^{\text{nx}} \\ 0 & \dots & 0 & 0 & 1 & 0 & \dots & \dots & 0 & \beta & \alpha & \beta & 0 & \dots \\ \vdots & \dots & \vdots & \vdots & 0 & 1 & 0 & \dots & \vdots & 0 & \beta & \alpha & \beta & 0 \\ \vdots & \dots & \vdots & \vdots & \ddots & \ddots & \ddots & \ddots & \vdots & \vdots & \ddots & \ddots & \ddots & \ddots \\ 0 & \dots & 0 & 0 & \dots & \dots & 0 & 1 & 0 & \beta & \dots & 0 & \beta & \alpha \end{bmatrix}.$$

$\boldsymbol{S^4_{nx\times[(ny-1)nx]}}$ stands for the coefficients for the discretization of the vertical velocity at the bottom and top of the domain. $\boldsymbol{S^5_{[nx(ny-2)]\times[(ny-1)nx]}}$ represents the last columns and inner rows that correspond to the coefficients for the discretization of the vertical velocity at inner nodes and $\boldsymbol{S^6_{[(ny-1)nx]\times N}}$ consists of only zero entries corresponding to the discretization for the vertical velocity:

$$\boldsymbol{S^4} = \begin{bmatrix} \overbrace{0 & \dots & 0}^{\text{(ny-1)nx}} \\ \vdots & \dots & \vdots \\ \vdots & \dots & \vdots \\ 0 & \dots & 0 \end{bmatrix}, \quad \boldsymbol{S^5} = \begin{bmatrix} \overbrace{0 & \dots & 0}^{\text{(ny-1)nx}} \\ \vdots & \dots & \vdots \\ \vdots & \dots & \vdots \\ 0 & \dots & 0 \end{bmatrix}. \tag{5.16}$$

The pressure vector with two indices p_{ij} can be mapped into the single index velocity vector notation p_k:

$$p_{ij} = p_{(j-2)(nx)+(i-1)} \quad \text{for} \quad j = 2, ..., ny+1, \quad i = 2, ..., nx+1. \tag{5.17}$$

The boundary condition for the pressure with linear interpolation is given by

$$\frac{p_{0,j} + p_{1,j}}{2} = \frac{p_{nx+1,j} + p_{nx+2,j}}{2} + c \quad \rightarrow \quad p_{0,j} = p_{nx+2,j} + p_{nx+1,j} - p_{1,j} + 2c. \tag{5.18}$$

The pressure gradient is also equal at the inlet and outlet boundaries:

$$\frac{p_{1,j} - p_{0,j}}{h_1} = \frac{p_{nx+2,j} - p_{nx+1,j}}{h_1} \quad \rightarrow \quad p_{0,j} = p_{1,j} + p_{nx+1,j} - p_{nx+2,j}. \tag{5.19}$$

The following relation (periodic boundary condition) can be obtained using the information about the ghost pressure node $p_{0,j}$ from Eq. (5.18):

$$p_{0,j} = p_{nx+2,j} + p_{nx+1,j} - p_{1,j} + 2c = p_{1,j} + p_{nx+1,j} - p_{nx+2,j}, \tag{5.20}$$

$$p_{nx+2,j} = p_{1,j} - c. \tag{5.21}$$

The ghost value at grid cells $(nx+2), j$ can be replaced by known values. The ghost value $p_{0,j}$ can also be replaced by known values using periodicity

$$p_{0,j} = p_{1,j} + p_{nx+1,j} - p_{nx+2,j} = p_{1,j} + p_{nx+1,j} - (p_{1,j} - c) = p_{nx+1,j} + c. \tag{5.22}$$

The size of the pressure vector $\boldsymbol{p}$ is $[(nx)(ny)] \times 1$. The linear algebraic operator $\boldsymbol{G}$ representing the discretization of the projection of the gradient in the direction of the axis

of the cylinder is

$$\boldsymbol{G}_{\boldsymbol{N}\times[(\boldsymbol{nx})(\boldsymbol{ny})]} = \frac{1}{h_1}\begin{bmatrix} \boldsymbol{G^1} & 0 & \dots & \dots & 0 \\ 0 & \boldsymbol{G^1} & 0 & \dots & \vdots \\ \vdots & \ddots & \ddots & \ddots & 0 \\ \vdots & \dots & 0 & \boldsymbol{G^1} & 0 \\ 0 & \dots & \dots & 0 & \boldsymbol{G^1} \\ 0 & \dots & \dots & \dots & 0 \\ \vdots & \dots & \dots & \dots & \vdots \\ 0 & \dots & \dots & \dots & 0 \end{bmatrix}.$$

Last $nx(ny-1)$ rows of matrix $\boldsymbol{G}$ have only zero entries since the pressure is assumed to be constant in vertical direction and first $ny(nx+1)$ rows are for the discretization in x_1 direction:

$$\boldsymbol{G}_{\boldsymbol{nx}\times\boldsymbol{nx}} = \begin{bmatrix} \overbrace{\begin{matrix} 1 & 0 & \dots & \dots & \dots \end{matrix}}^{\text{(nx)}} & -1 \\ -1 & 1 & 0 & \dots & \dots & 0 \\ 0 & -1 & 1 & 0 & \dots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & 0 & -1 & 1 & 0 \\ 0 & \dots & \dots & 0 & -1 & 1 \end{bmatrix}.$$

The matrix $\bar{\boldsymbol{G}}$ has non-zero entries only in the point connected to inflow and outflow boundary and has zero entries in last $nx(ny-1)$ rows. It can be represented with vector $\bar{\boldsymbol{G}}^{\boldsymbol{1}}$ as follows:

$$\bar{\boldsymbol{G}}_{\boldsymbol{N}\times\boldsymbol{1}} = \tfrac{1}{h_1}\begin{bmatrix} \bar{\boldsymbol{G}}^{\boldsymbol{1}} \\ \vdots \\ \bar{\boldsymbol{G}}^{\boldsymbol{1}} \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \quad \bar{\boldsymbol{G}}^{\boldsymbol{1}}_{\boldsymbol{nx}\times\boldsymbol{nx}} = \begin{bmatrix} -1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}.$$

5.2.1 Results and discussion

In order to solve equations (5.7)-(5.9), the pressure correction method described by [24] is extended to incorporate the unknown constant c and the extra equation (5.9).

As a first step toward the construction of an efficient numerical scheme, the following system of linear equations are obtained (based on the algorithm used in [25]) by discretizing Navier-Stokes equations in time using forward Euler (explicit) method (whereas a second

order accurate time integration method is used in the model for turbulent flow):

$$\boldsymbol{M}\frac{\boldsymbol{u}^{n+1}-\boldsymbol{u}^{n}}{\Delta t}+\boldsymbol{S}\boldsymbol{u}^{n}+\boldsymbol{G}\boldsymbol{p}^{n}+\bar{\boldsymbol{G}}c^{n}=0, \tag{5.23}$$

$$\boldsymbol{D}\boldsymbol{u}^{n+1}=\boldsymbol{0}, \tag{5.24}$$

$$\boldsymbol{R}\boldsymbol{u}^{n+1}=Q. \tag{5.25}$$

The constant c is treated as the pressure $\boldsymbol{p}$ in the approach for pressure correction. This usual approach suggests to replace $\boldsymbol{u}^{n+1}$ in Eq. (5.23) with a predictor $\boldsymbol{u}^{*}$:

$$\boldsymbol{M}\frac{\boldsymbol{u}^{*}-\boldsymbol{u}^{n}}{\Delta t}+\boldsymbol{S}\boldsymbol{u}^{n}+\boldsymbol{G}\boldsymbol{p}^{n}+\bar{\boldsymbol{G}}c^{n}=0. \tag{5.26}$$

The velocity field $\boldsymbol{u}^{*}$ needs to be projected on the space of divergence-free vector fields as well as on vector fields satisfying Eq. (5.9). Therefore, Eq. (5.26) is subtracted from Eq. (5.23):

$$\boldsymbol{M}\frac{\boldsymbol{u}^{n+1}-\boldsymbol{u}^{*}}{\Delta t}+\boldsymbol{G}(\boldsymbol{p}^{n+1}-\boldsymbol{p}^{n})+\bar{\boldsymbol{G}}(c^{n+1}-c^{n})=0. \tag{5.27}$$

In order to obtain a relation for pressure correction and constant c, Eq. (5.27) is pre-multiplied by $\boldsymbol{D}\boldsymbol{M}^{-1}$ and $\boldsymbol{R}\boldsymbol{M}^{-1}$ respectively:

$$\frac{\boldsymbol{D}\boldsymbol{u}^{n+1}-\boldsymbol{D}\boldsymbol{u}^{*}}{\Delta t}+\boldsymbol{D}\boldsymbol{M}^{-1}\boldsymbol{G}(\boldsymbol{p}^{n+1}-\boldsymbol{p}^{n})+\boldsymbol{D}\boldsymbol{M}^{-1}\bar{\boldsymbol{G}}(c^{n+1}-c^{n})=0, \tag{5.28}$$

$$\frac{\boldsymbol{R}\boldsymbol{u}^{n+1}-\boldsymbol{R}\boldsymbol{u}^{*}}{\Delta t}+\boldsymbol{R}\boldsymbol{M}^{-1}\boldsymbol{G}(\boldsymbol{p}^{n+1}-\boldsymbol{p}^{n})+\boldsymbol{R}\boldsymbol{M}^{-1}\bar{\boldsymbol{G}}(c^{n+1}-c^{n})=0, \tag{5.29}$$

and the following equations are obtained by applying Eqs.(5.24) and (5.25):

$$\frac{-\boldsymbol{D}\boldsymbol{u}^{*}}{\Delta t}+\boldsymbol{D}\boldsymbol{M}^{-1}\boldsymbol{G}(\boldsymbol{p}^{n+1}-\boldsymbol{p}^{n})+\boldsymbol{D}\boldsymbol{M}^{-1}\bar{\boldsymbol{G}}(c^{n+1}-c^{n})=0, \tag{5.30}$$

$$\frac{Q-\boldsymbol{R}\boldsymbol{u}^{*}}{\Delta t}+\boldsymbol{R}\boldsymbol{M}^{-1}\boldsymbol{G}(\boldsymbol{p}^{n+1}-\boldsymbol{p}^{n})+\boldsymbol{R}\boldsymbol{M}^{-1}\bar{\boldsymbol{G}}(c^{n+1}-c^{n})=0. \tag{5.31}$$

Since the result of $\boldsymbol{R}\boldsymbol{M}^{-1}\bar{\boldsymbol{G}}$ is a number, the equation for the constant c can be represented as follows:

$$c^{n+1}-c^{n}=\frac{-1}{\boldsymbol{R}\boldsymbol{M}^{-1}\bar{\boldsymbol{G}}}\left(\boldsymbol{R}\boldsymbol{M}^{-1}\boldsymbol{G}(p^{n+1}-p^{n})+\frac{Q-\boldsymbol{R}\boldsymbol{u}^{*}}{\Delta t}\right). \tag{5.32}$$

In order to compute the pressure correction, the following modified Laplacian type equation is defined

$$\boldsymbol{D}\boldsymbol{M}^{-1}\left(\boldsymbol{G}-\frac{\bar{\boldsymbol{G}}\boldsymbol{R}\boldsymbol{M}^{-1}\boldsymbol{G}}{\boldsymbol{R}\boldsymbol{M}^{-1}\bar{\boldsymbol{G}}}\right)(p^{n+1}-p^{n})=\frac{\boldsymbol{D}\boldsymbol{u}^{*}}{\Delta t}+\frac{\boldsymbol{D}\boldsymbol{M}^{-1}\bar{\boldsymbol{G}}}{\boldsymbol{R}\boldsymbol{M}^{-1}\bar{\boldsymbol{G}}}\frac{Q-\boldsymbol{R}\boldsymbol{u}^{*}}{\Delta t}. \tag{5.33}$$

After calculating pressure and constant c corrections, $\boldsymbol{u}^{n+1}$ is computed from Eq. (5.27):

$$\boldsymbol{u}^{n+1}=\boldsymbol{u}^{*}-\boldsymbol{M}^{-1}\Delta t\left[\boldsymbol{G}(\boldsymbol{p}^{n+1}-\boldsymbol{p}^{n})+\bar{\boldsymbol{G}}(c^{n+1}-c^{n})\right]. \tag{5.34}$$

If the prescribed flow rate Q is assumed to be equal to the current flow rate $Q = \boldsymbol{Ru}^*$ (Hagen-Poiseuille flow with fully developed velocity profile) and the flow is assumed to be divergence-free, then the pressure correction equation becomes

$$\boldsymbol{DM}^{-1}\left(\boldsymbol{G} - \frac{\bar{\boldsymbol{G}}\boldsymbol{R}\boldsymbol{M}^{-1}\boldsymbol{G}}{\boldsymbol{R}\boldsymbol{M}^{-1}\bar{\boldsymbol{G}}}\right)(p^{n+1} - p^n) = \frac{\cancelto{0}{\boldsymbol{Du}^*}}{\Delta t} + \frac{\boldsymbol{D}\boldsymbol{M}^{-1}\bar{\boldsymbol{G}}}{\boldsymbol{R}\boldsymbol{M}^{-1}\bar{\boldsymbol{G}}}\frac{\cancelto{0}{Q - \boldsymbol{Ru}^*}}{\Delta t}, \tag{5.35}$$

which yields the below equation for the pressure at next time step:

$$p^{n+1} = p^n. \tag{5.36}$$

Eq. (5.36) results in the following relation for the constant c:

$$c^{n+1} - c^n = \frac{-1}{\boldsymbol{R}\boldsymbol{M}^{-1}\bar{\boldsymbol{G}}}\left(\boldsymbol{R}\boldsymbol{M}^{-1}\boldsymbol{G}\cancelto{0}{(p^{n+1} - p^n)} + \frac{\cancelto{0}{Q - \boldsymbol{Ru}^*}}{\Delta t}\right), \tag{5.37}$$

$$c^{n+1} = c^n. \tag{5.38}$$

In order to test the method described above, a channel flow of water is considered with necessary boundary conditions. The height and length of the channel are $H = 0.05\ m$ and $L = 0.25\ m$ respectively. The Reynolds number for the centre line velocity (u_c) of this steady state flow is $Re_c = 997$. The steady-state fully developed Hagen-Poiseuille channel flow is simulated long enough to obtain converged results for both pressure drop and velocity. Figures 5.4 and 5.5 represent the contour for the velocity and pressure respectively in the computational domain. Figures 5.6 and 5.7 show the velocity profile and the pressure drop for the flow rate $Q = \boldsymbol{Ru}^* = 0.001\ m^2\ s^{-1}$. The theoretical values

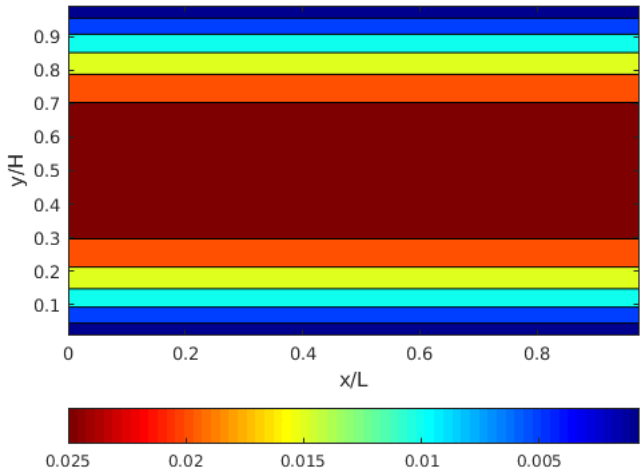


Figure 5.4: Velocity contour for a channel flow with $Re_c = 997$

of the constant pressure gradient force, which is pressure drop divided by the channel

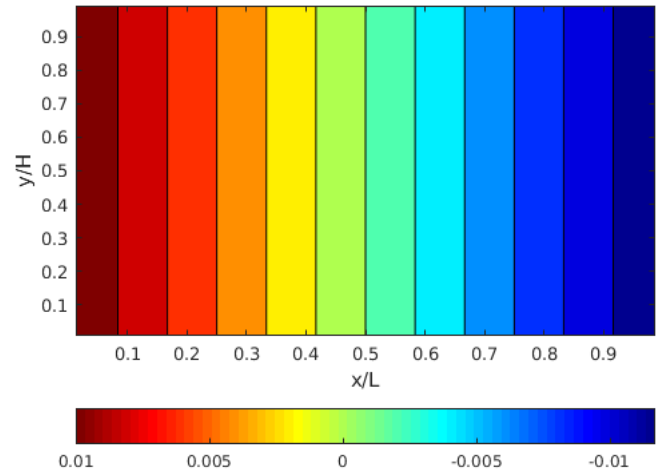


Figure 5.5: Pressure contour for a channel flow with $Re_c = 997$

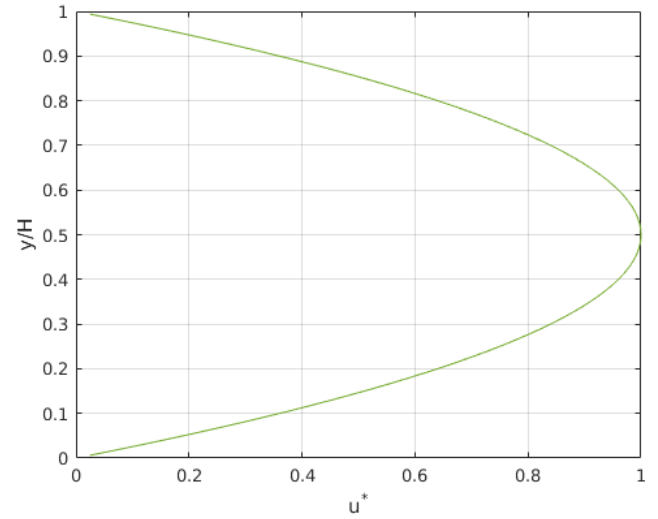


Figure 5.6: Dimensionalized velocity profile ($u^* = u/u_c$) with respect to the center line velocity for $Re_c = 997$

length, and for the center line velocity are $9.6 \cdot 10^{-2}\ kg\ m^{-2}\ s^{-2}$ and $3 \cdot 10^{-2}\ m\ s^{-2}$ respectively. The computed numerical results are exactly the same as the theoretical results.

Initially, the viscosity is assumed to be constant for simplicity. However, for the later stage the equations are extended to allow for a non-constant viscosity formulation. The numerical results of the simulation with constant viscosity match exactly with the theoretical results that are calculated using the Hagen-Poiseuille equation. For a prescribed flow rate different than the actual one, the average velocity and pressure gradient increase until they converge to the desired values.

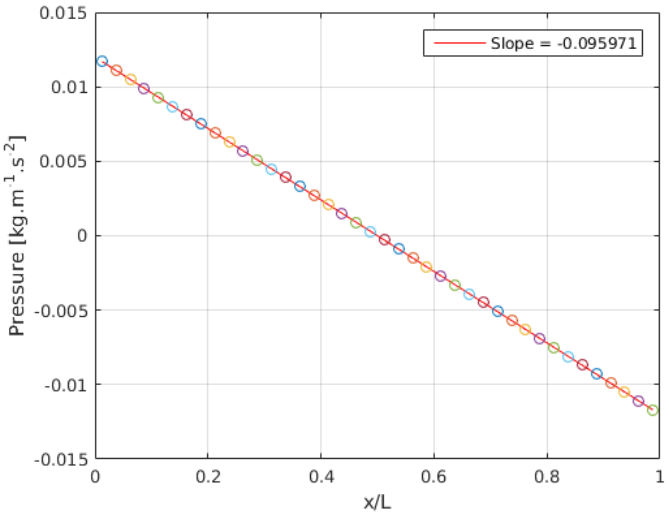


Figure 5.7: Pressure drop along the x_1 direction for $Re_c = 997$

Figures 5.8 and 5.9 represent the contour for the velocity and pressure respectively in the computational domain and Figures 5.10 and 5.11 stand for the velocity profile and the pressure drop for the flow rate $Q = 0.002\ m^2\ s^{-1}$, in which the flow is not fully developed since the velocity profile changes with respect to time until it reaches the steady state solution. Further to this transient behavior, the flow will become steady and fully developed and the velocity profile no longer changes in time. The Reynolds number of this steady state flow is $Re_c = 1994$. The theoretical values of the constant pressure gradient force and the maximum velocity at the center line are $1.92 \cdot 10^{-1}\ kg\ m^{-2}\ s^{-2}$ and $6.0 \cdot 10^{-1}\ m\ s^{-2}$ respectively for the given flow rate. The computed numerical results converge exactly to these theoretical results. For these simulations, the number of grid

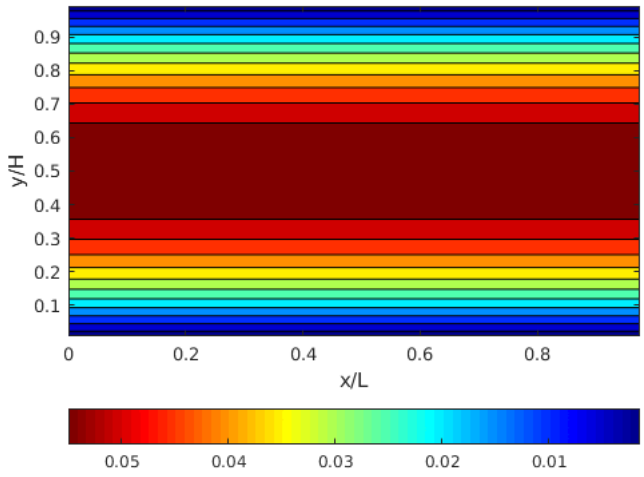


Figure 5.8: Velocity contour for a channel flow with $Re_c = 1994$

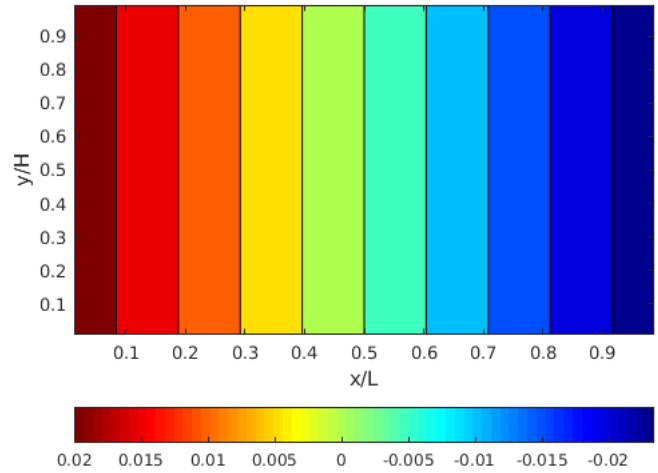


Figure 5.9: Pressure contour for a channel flow with $Re_c = 1994$

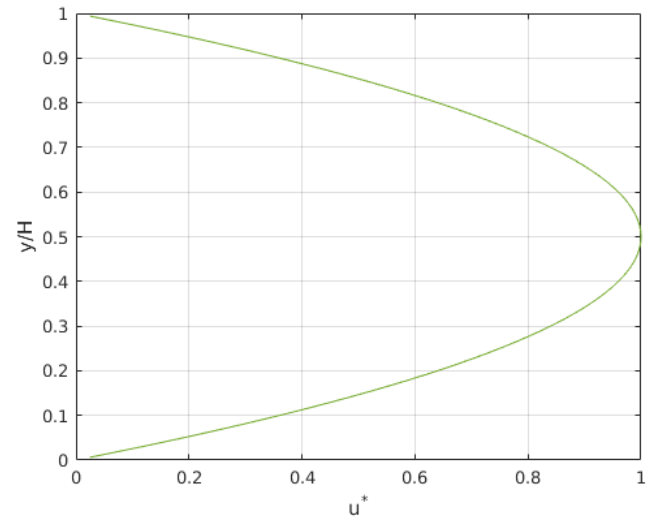


Figure 5.10: Dimensionalized velocity profile ($u^* = u/u_c$) with respect to the center line velocity for $Re_c = 1994$

points in the x_1-direction is kept constant and equal to $nx = 40$, and the number of grid points in the x_2-direction is $ny = 3^4 = 81$. Since the pressure values are located at the cell centers, the first and last pressure values in Fig. 5.7 are not located at the inlet and outlet of the channel.

The pressure correction equation is iteratively solved until the specified convergence criteria for the pressure and velocity values are achieved. The number of sample points is fixed and the solution on grids, where the mesh width is refined with a factor of 3, are used for error comparison. This means that the same points are present in the finer meshes. Thus, the solution at those points can be compared to check convergence. The L^2-norm

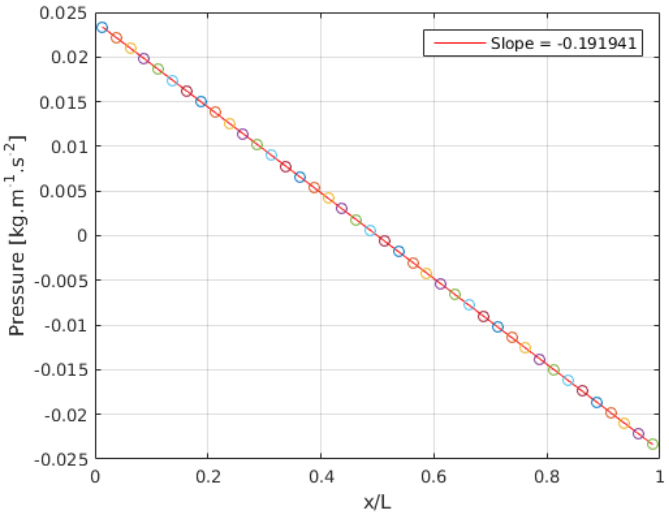
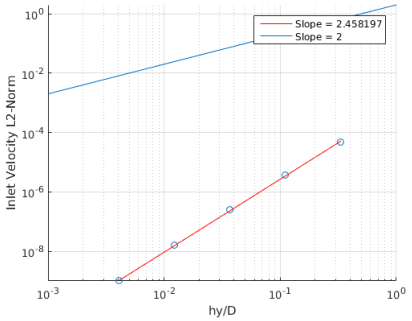
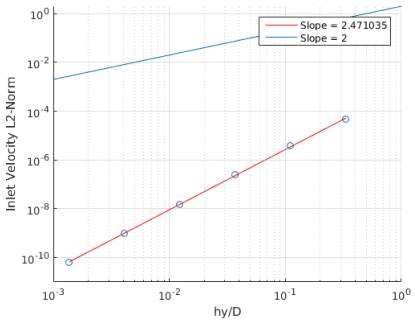


Figure 5.11: Pressure drop along the x_1 direction for $Re_c = 1994$

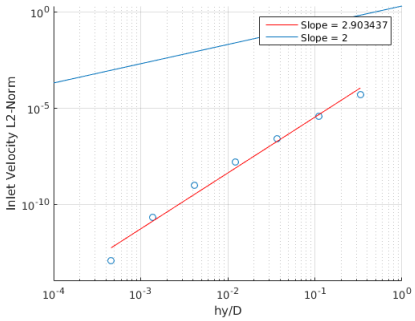
of the error between the numerical and theoretical solution of the pressure gradient is plotted with respect to the dimensionless parameter h_2/H, where h_2 is the grid spacing in direction x_2. Figures 5.12 and 5.13 represent the error between theoretical and numer-



(a) For $ny = \{3^1, 3^2, ..., 3^5 = 3, 9, ..., 243\}$



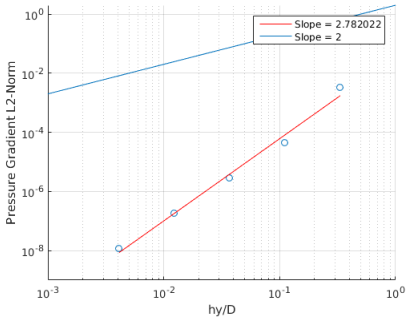
(b) For $ny = \{3^1, 3^2, ..., 3^6 = 3, 9, ...729\}$



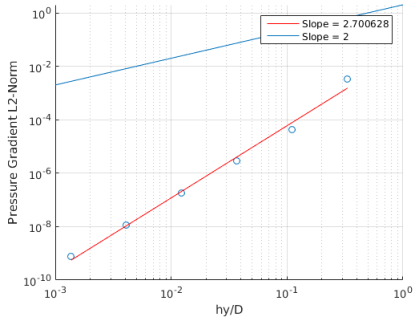
(c) For $ny = \{3^1, 3^2, ..., 3^7 = 3, 9, ..., 2187\}$

Figure 5.12: L_2 norm error of the inflow velocity field for three different configurations

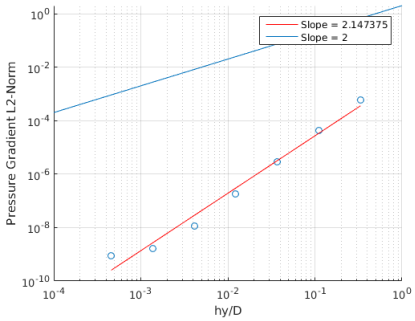
ical solutions of inflow velocity field and pressure drop respectively. In Fig. 5.12a, the error of the inflow velocity field is computed for five different mesh width configurations; i.e. the number of grid points in the vertical direction are 3, 9, 27, 81, 243 accordingly. Figures 5.12b and 5.12c illustrate the order of magnitude of the error (second-order) for six and seven different solutions simulated with different mesh widths respectively. The error of the pressure drop is computed in Figures 5.13a, 5.13b and 5.13c for five, six, and seven different mesh width configurations to identify the asymptotic region and the region where results were affected by round-off error. The number of grid points changes only in the vertical direction in each simulation to be able to compute the convergence of inlet velocity and pressure. It has been demonstrated that this algorithm for the solution of



(a) For $ny = \{3^1, 3^2, ..., 3^5 = 3, 9, ..., 243\}$



(b) For $ny = \{3^1, 3^2, ..., 3^6 = 3, 9, ..., 729\}$



(c) For $ny = \{3^1, 3^2, ..., 3^7 = 3, 9, ..., 2187\}$

Figure 5.13: L_2 norm error of the pressure for three different configurations

the incompressible Navier-Stokes equations converges fast and is well suited for modeling periodic boundary conditions in combination with a prescribed flow rate and an unknown jump in the pressure.

The following system of linear equations are obtained by discretizing the Navier-Stokes equations in time using backward Euler (implicit) method:

$$\boldsymbol{M}\frac{\boldsymbol{u}^{n+1}-\boldsymbol{u}^{n}}{\Delta t}+\boldsymbol{S}\boldsymbol{u}^{n+1}+\boldsymbol{G}\boldsymbol{p}^{n}+\bar{\boldsymbol{G}}c^{n}=0. \tag{5.39}$$

The velocity at the next time step is unknown. Therefore, it is predicted by following the pressure correction approach and replacing $\boldsymbol{u}^{n+1}$ in Eq. (5.39) by a predictor velocity $\boldsymbol{u}^{*}$:

$$\boldsymbol{M}\frac{\boldsymbol{u}^{*}-\boldsymbol{u}^{n}}{\Delta t}+\boldsymbol{S}\boldsymbol{u}^{*}+\boldsymbol{G}\boldsymbol{p}^{n}+\bar{\boldsymbol{G}}c^{n}=0. \tag{5.40}$$

A new term appears in the predictor equation for the implicit method which results in the following equation:

$$\boldsymbol{u}^* = \boldsymbol{u}^n - \boldsymbol{M}^{-1}\Delta t(\boldsymbol{S}\boldsymbol{u}^* + \boldsymbol{G}\boldsymbol{p}^n + \bar{\boldsymbol{G}}c^n), \tag{5.41}$$

and the predictor velocity is gathered on the left hand side:

$$(\mathbb{1} + \boldsymbol{M}^{-1}\Delta t\boldsymbol{S})\boldsymbol{u}^* = \boldsymbol{u}^n - \boldsymbol{M}^{-1}\Delta t(\boldsymbol{G}\boldsymbol{p}^n + \bar{\boldsymbol{G}}c^n). \tag{5.42}$$

After taking the predictor velocity to the left hand side, the following equation is obtained:

$$\boldsymbol{u}^* = (\mathbb{1} + \boldsymbol{M}^{-1}\Delta t\boldsymbol{S})^{-1}[\boldsymbol{u}^n - \boldsymbol{M}^{-1}\Delta t(\boldsymbol{G}\boldsymbol{p}^n + \bar{\boldsymbol{G}}c^n)]. \tag{5.43}$$

In order to correct the velocity field, $\boldsymbol{u}^*$ has to be projected on the necessary fields by subtracting Eq. (5.40) from Eq. (5.39). This yields:

$$\boldsymbol{M}\frac{\boldsymbol{u}^{n+1} - \boldsymbol{u}^*}{\Delta t} + \boldsymbol{G}(\boldsymbol{p}^{n+1} - \boldsymbol{p}^n) + \bar{\boldsymbol{G}}(c^{n+1} - c^n) = 0. \tag{5.44}$$

This implicit algorithm for the solution of the incompressible Navier-Stokes equations converges faster than the explicit one. The convergence of the constant c shows exactly the same behavior as the convergence of the pressure to reach the stationary state. Fig. 5.14 shows the relative error of c (the difference between the theoretical value and the computed c) as a function of number of iterations.

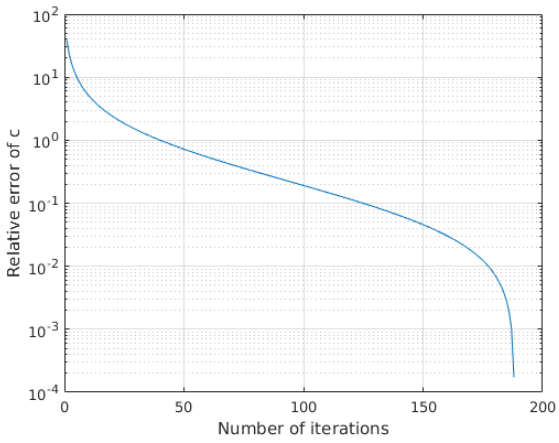


Figure 5.14: Relative error of c as a function of the iteration number

5.3 2D Poiseuille axisymmetric pipe flow

The boundary conditions for the velocity and pressure in a pipe are given by

$$\boldsymbol{u}(r,\theta,z)\Big|_{inlet} = \boldsymbol{u}(r,\theta,z+L)\Big|_{outlet}, \tag{5.45}$$

$$\frac{\partial \boldsymbol{u}(r,\theta,z)}{\partial n}|_{inlet} = -\frac{\partial \boldsymbol{u}(r,\theta,z+L)}{\partial n}|_{outlet}, \tag{5.46}$$

$$\boldsymbol{p}(r,\theta,z)_{inlet} = \boldsymbol{p}(r,\theta,z+L)_{outlet} + c. \tag{5.47}$$

where r, θ and z represent the radial, circumferential and axial directions respectively, L denotes the pipe length, $\boldsymbol{u}$ is the algebraic vector (in $3D$), $\boldsymbol{p}$ is the pressure vector, $\boldsymbol{n}$ is the outward directed normal and c represents the unknown constant jump in pressure, which stands for the periodicity of the pressure.

The continuity equations for an incompressible flow in cylindrical coordinates can be simplified as follows:

$$\frac{\partial u_z}{\partial z} + \cancelto{0}{\frac{\partial u_r}{\partial r}} + \cancel{\frac{1}{r}}\cancelto{0}{\frac{\partial u_\theta}{\partial \theta}} + \cancelto{0}{\frac{u_r}{r}} = 0, \tag{5.48}$$

where u_z, u_r, and u_θ are velocity components of axial, radial, and azimuthal directions respectively in cylindrical coordinates. The above simplifications are possible since there assumed to exist a flow only in the axial direction. For a fully developed Hagen-Poiseuille incompressible flow with a variable viscosity, the momentum equations written in cylindrical coordinates can be simplified in axial, radial and circumferential directions respectively as follows:

$$\begin{aligned}\frac{\partial u_z}{\partial t} + \cancelto{0}{\frac{\partial u_z u_z}{\partial z}} + \cancelto{0}{\frac{\partial u_r u_z}{\partial r}} + \cancel{\frac{1}{r}}\cancelto{0}{\frac{\partial u_\theta u_z}{\partial \theta}} + \cancelto{0}{\frac{u_r u_z}{r}} + \frac{1}{\rho}\frac{\partial p}{\partial z} \\ = \cancelto{0}{\frac{\partial \tau_{zz}}{\partial z}} + \frac{\partial \tau_{rz}}{\partial r} + \cancel{\frac{1}{r}}\cancelto{0}{\frac{\partial \tau_{\theta z}}{\partial \theta}} + \frac{\tau_{rz}}{r} + \cancelto{0}{f_z},\end{aligned} \tag{5.49}$$

$$\begin{aligned}\cancelto{0}{\frac{\partial u_r}{\partial t}} + \cancelto{0}{\frac{\partial u_z u_r}{\partial z}} + \cancelto{0}{\frac{\partial u_r u_r}{\partial r}} + \cancel{\frac{1}{r}}\cancelto{0}{\frac{\partial u_\theta u_r}{\partial \theta}} + \cancelto{0}{\frac{(u_r u_r - u_\theta u_\theta)}{r}} + \cancelto{0}{\frac{1}{\rho}\frac{\partial p}{\partial r}} \\ = \cancelto{0}{\frac{\partial \tau_{zr}}{\partial z}} + \cancelto{0}{\frac{\partial \tau_{rr}}{\partial r}} + \cancel{\frac{1}{r}}\cancelto{0}{\frac{\partial \tau_{\theta r}}{\partial \theta}} + \cancelto{0}{\frac{(\tau_{rr} - \tau_{\theta\theta})}{r}} + \cancelto{0}{f_r},\end{aligned} \tag{5.50}$$

$$\begin{aligned}\cancelto{0}{\frac{\partial u_\theta}{\partial t}} + \cancelto{0}{\frac{\partial u_z u_\theta}{\partial z}} + \cancelto{0}{\frac{\partial u_r u_\theta}{\partial r}} + \cancel{\frac{1}{r}}\cancelto{0}{\frac{\partial u_\theta u_\theta}{\partial \theta}} + 2\cancelto{0}{\frac{u_\theta u_\theta}{r}} + \cancelto{0}{\frac{1}{\rho}\frac{1}{r}\frac{\partial p}{\partial \theta}} \\ = \cancelto{0}{\frac{\partial \tau_{z\theta}}{\partial z}} + \cancelto{0}{\frac{\partial \tau_{r\theta}}{\partial r}} + \cancel{\frac{1}{r}}\cancelto{0}{\frac{\partial \tau_{\theta\theta}}{\partial \theta}} + 2\cancelto{0}{\frac{\tau_{r\theta}}{r}} + \cancelto{0}{f_\theta},\end{aligned} \tag{5.51}$$

where ρ is the density, p is the pressure term, f_z, f_r, and f_θ are body force components (all are equal to zero). Since the only varying velocity component is the axial velocity, which varies in the radial direction, the above simplifications are carried out. The coefficients of

the viscous stress tensor τ_{ij} $(i,j = z,r,\theta)$ for a Newtonian fluid are given by:

$$\tau_{zz} = 2\nu\left(\cancelto{0}{\frac{\partial u_z}{\partial z}}\right), \quad \tau_{zr} = \tau_{rz} = \nu\left(\frac{\partial u_z}{\partial r} + \cancelto{0}{\frac{\partial u_r}{\partial z}}\right),$$

$$\tau_{rr} = 2\nu\cancelto{0}{\left(\frac{\partial u_r}{\partial r}\right)}, \quad \tau_{\theta\theta} = 2\nu\cancelto{0}{\left(\frac{1}{r}\frac{\partial u_\theta}{\partial \theta} + \frac{u_r}{r}\right)},$$

$$\tau_{z\theta} = \tau_{\theta z} = \nu\cancelto{0}{\left(\frac{\partial u_\theta}{\partial z} + \frac{1}{r}\frac{\partial u_z}{\partial \theta}\right)}, \quad \tau_{r\theta} = \tau_{\theta r} = \nu\cancelto{0}{\left(\frac{\partial u_\theta}{\partial r} - \frac{u_\theta}{r} + \frac{1}{r}\frac{\partial u_r}{\partial r}\right)}.$$

Eq. (5.48) implies the τ_{zz} stress is equal to zero. As a result of these simplifications, the following system of equations are obtained:

$$\frac{\partial u_z}{\partial z} = 0, \tag{5.52}$$

$$\frac{\partial u_z}{\partial t} + \frac{1}{\rho}\frac{\partial p}{\partial z} = \frac{1}{\rho}\frac{\partial}{\partial r}\left(\mu\frac{\partial u_z}{\partial r}\right) + \frac{1}{\rho}\frac{\mu}{r}\frac{\partial u_z}{\partial r} = \frac{1}{\rho}\frac{1}{r}\frac{\partial}{\partial r}\left(r\mu\frac{\partial u_z}{\partial r}\right). \tag{5.53}$$

The only difference between the channel and pipe flow is the scaling of the viscous term involving the radius. For illustration purposes the discretization of the viscous term is only carried out in the radial direction with index j:

$$\left[\frac{1}{\rho}\frac{1}{r_j}\frac{\partial}{\partial r_j}\left(r\mu\frac{\partial u_z}{\partial r}\right)\right]\Big|_j = \frac{1}{\rho}\frac{1}{r_j}\frac{\left(r\mu\frac{\partial u_z}{\partial r}\right)|_{j+\frac{1}{2}} - \left(r\mu\frac{\partial u_z}{\partial r}\right)|_{j-\frac{1}{2}}}{\Delta r_j^2}. \tag{5.54}$$

The discretization at $j = \frac{3}{2}$:

$$\frac{1}{\rho}\frac{1}{r_{\frac{3}{2}}}\frac{\left(r\mu\frac{\partial u_z}{\partial r}\right)|_{j=2} - \left(r\mu\frac{\partial u_z}{\partial r}\right)|_{j=1}}{\Delta r_{\frac{3}{2}}^2} = \frac{1}{\rho}\frac{1}{r_{\frac{3}{2}}}\frac{\left[r_2\mu_2\left((u_z)_{\frac{5}{2}} - (u_z)_{\frac{3}{2}}\right)\right]}{\Delta r_{\frac{3}{2}}^2}, \tag{5.55}$$

where $r_1 = 0$. Thus, the velocity and pressure nodes are located at $(j + \frac{1}{2})$, in which $j = 1, 2...Nr/2$ and Nr is the number of total grid points in the radial direction.

5.3.1 Results and discussion

In order to test the method described above, a pipe flow of water is considered with necessary boundary conditions. The diameter and length of the pipe are $D = 0.05\ m$ and $L = 0.25\ m$ respectively. The Reynolds number for the center line velocity is 997. The steady-state fully developed Hagen-Poiseuille pipe flow is simulated long enough to obtain converged results for both pressure drop and velocity.

Figures 5.15 and 5.16 show the velocity and pressure contour. Figures 5.17 and 5.18 represent the velocity profile and the pressure drop for the upper half of the pipe wtih

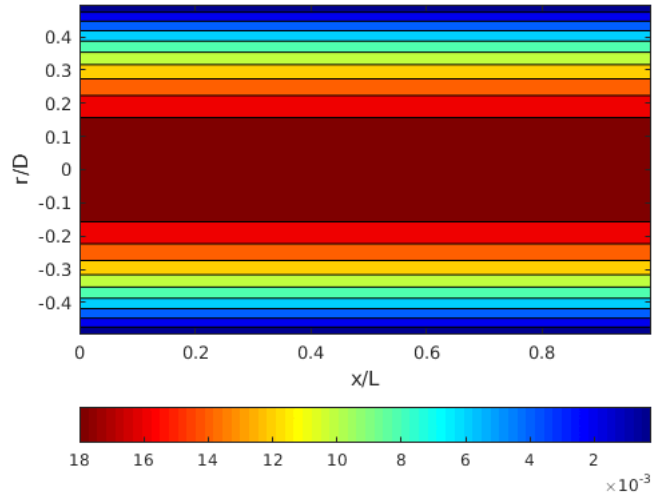


Figure 5.15: Velocity contour for a channel flow with $Re_c = 997$

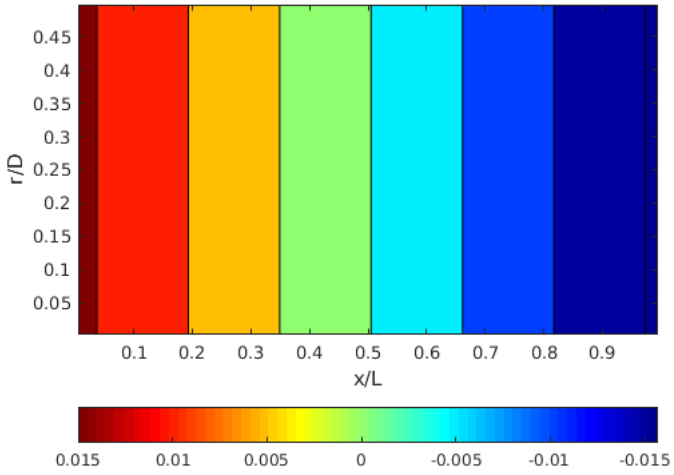


Figure 5.16: Pressure contour for a channel flow with $Re_c = 997$

the prescribed flow rate $Q = \boldsymbol{Ru}^* = 6 \cdot 10^{-4}\ m^2\ s^{-1}$. Fig. 5.19 illustrates the relative error of c (the difference between the theoretical value and the computed c) as a function of number of iterations for the Poiseuille axisymmetric pipe flow.

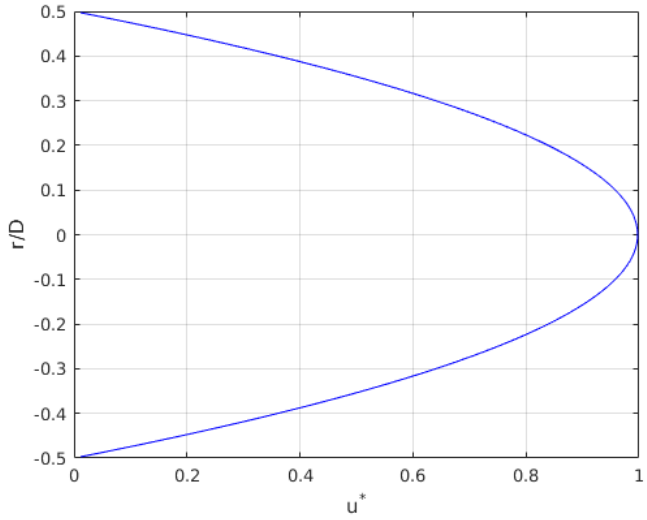


Figure 5.17: Dimensionalized velocity profile ($u^* = u/u_{th,c}$) with respect to the theoretical center line velocity for $Re_c = 997$

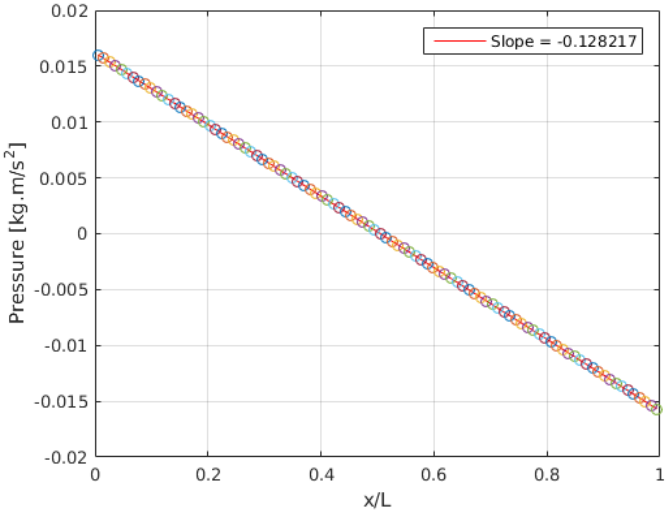


Figure 5.18: Pressure drop along the x_1 direction for $Re_c = 997$

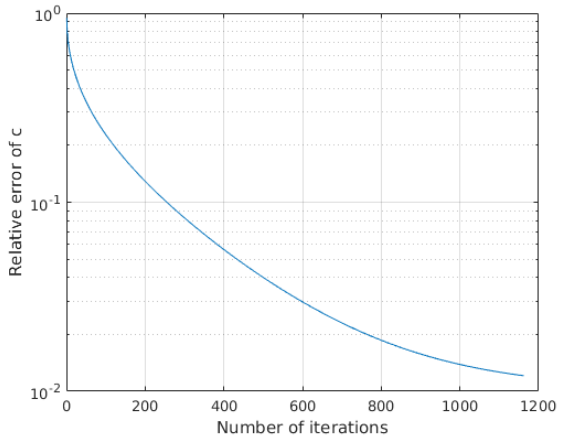


Figure 5.19: Relative error of c as a function of the iteration number

5.4 3D Poiseuille axisymmetric pipe flow

The necessary modifications are incorporated in the 3D MCLS algorithm. The code is modified in order to calculate the unknown constant c and simultaneously apply it as a boundary condition for pressure. Moreover, the right hand side and the left hand side matrices for the Poisson equation are modified such that Eq. (5.33) is satisfied. The simulations were done for a water flow on a grid of $N_r \times N_t \times N_z$ [3] cells until the given tolerance for the volumetric flow rate is achieved $|Q - \boldsymbol{R}\,\boldsymbol{u}*\,|/Q$. $\boldsymbol{Ru}*$ represents the initial volumetric flow rate of the flow. Table 5.2 indicates the results of runs for different

case	grid	Δt $[s]$	t_{total} $[s]$	Q $[m^3 \cdot s^{-1}]$	**Ru*** $[m^3 \cdot s^{-1}]$
1	$8 \times 10 \times 4$	$1 \cdot 10^{-1}$	0.3	$2 \cdot 10^{-5}$	$9.89 \cdot 10^{-5}$
2	$8 \times 10 \times 4$	$1 \cdot 10^{-1}$	0.3	$5.5 \cdot 10^{-5}$	$9.89 \cdot 10^{-5}$
3	$8 \times 10 \times 4$	$1 \cdot 10^{-1}$	0.3	$1 \cdot 10^{-4}$	$9.89 \cdot 10^{-5}$
4	$8 \times 10 \times 4$	$1 \cdot 10^{-2}$	0.03	$1 \cdot 10^{-4}$	$9.89 \cdot 10^{-5}$
5	$40 \times 10 \times 4$	$1 \cdot 10^{-2}$	0.12	$1 \cdot 10^{-4}$	$9.82 \cdot 10^{-5}$
6	$40 \times 10 \times 24$	$1 \cdot 10^{-2}$	0.12	$1 \cdot 10^{-4}$	$9.82 \cdot 10^{-5}$
7	$50 \times 15 \times 28$	$1 \cdot 10^{-2}$	0.15	$1 \cdot 10^{-4}$	$9.81 \cdot 10^{-5}$
8	$50 \times 30 \times 4$	$1 \cdot 10^{-1}$	2.3	$1 \cdot 10^{-4}$	$9.81 \cdot 10^{-5}$
9	$60 \times 20 \times 4$	$1 \cdot 10^{-1}$	2.3	$1 \cdot 10^{-4}$	$9.81 \cdot 10^{-5}$
10	$70 \times 10 \times 18$	$1 \cdot 10^{-2}$	0.15	$1 \cdot 10^{-4}$	$9.81 \cdot 10^{-5}$

Table 5.1: Input parameters of different cases for laminar Poiseuille axisymmetric pipe flow

cases. The unknown pressure jump constant c, the pressure drop over the length of the pipe $\Delta p/L$ and the maximum velocity at the center of the pipe U_{max} are calculated and compared for different cases. The convergence of the volumetric flow rate is used as a stopping criterion.

[3] N_r, N_t and N_z represent the number of cells in radial, azimuthal and axial directions respectively.

The numerical results match with the theoretical results exactly as the number of grid points increase. The theoretical pressure drop over the length of the pipe and maximum velocity at the center of the pipe for Poiseuille flow are given by

$$\frac{\Delta p}{L} = -\frac{8\mu Q}{\pi r^4}, \tag{5.56}$$

$$U_z = -\frac{1}{4\mu}\frac{\partial p}{\partial z}(r^2 - \delta^2), \tag{5.57}$$

$$U_{max} = -\frac{1}{4\mu}\frac{\partial p}{\partial z}r^2, \tag{5.58}$$

where L is the length of the pipe, r is the pipe radius, δ represents the distance of the location from the pipe center, Q is the volumetric flow rate and μ is the dynamic viscosity. The theoretical values for the pressure gradient and the maximum velocity are calculated

case	Re_c	c $[N \cdot m^{-2}]$	$\frac{\Delta p}{L}$ $[N \cdot m^{-3}]$	u_c $[m \cdot s^{-1}]$	$\frac{\lvert Q - \mathbf{Ru^*}\rvert}{Q}$
1	997	-0.0323	-0.1294	$2.0030 \cdot 10^{-2}$	$2.875 \cdot 10^{-7}$
2	2492.5	-0.1119	-0.4476	$5.2666 \cdot 10^{-2}$	$8.144 \cdot 10^{-6}$
3	4985	-0.1617	-0.6470	$1.0014 \cdot 10^{-1}$	$2.875 \cdot 10^{-7}$
4	4985	-0.1618	-0.6475	$1.0014 \cdot 10^{-1}$	$3.171 \cdot 10^{-9}$
5	4985	-0.1947	-0.7791	$1.0092 \cdot 10^{-1}$	$9.772 \cdot 10^{-7}$
6	4985	-0.1947	-0.7791	$1.0092 \cdot 10^{-1}$	$9.772 \cdot 10^{-7}$
7	4985	-0.1914	-0.7658	$1.0093 \cdot 10^{-1}$	$9.628 \cdot 10^{-7}$
8	4985	-0.1681	-0.6725	$1.0104 \cdot 10^{-1}$	$2.0409 \cdot 10^{-7}$
9	4985	-0.1680	-0.6722	$1.0105 \cdot 10^{-1}$	$9.824 \cdot 10^{-7}$
10	4985	-0.1898	-0.7592	$1.0095 \cdot 10^{-1}$	$9.373 \cdot 10^{-7}$

Table 5.2: Resulting flow parameters of the cases, i.e., the pressure drop, the center line velocity and the relative error in the flow rate

for a single phase water flow with a flow rate of $1 \cdot 10^{-4}[m \cdot s^{-1}]$:

$$\frac{\Delta p}{L} = -\frac{(8 \cdot 10^{-3})(1 \cdot 10^{-4})}{\pi(0.025)^4} = -0.6518\ N \cdot m^{-3}, \tag{5.59}$$

$$u_c = -\frac{1}{4 \cdot 10^{-3}}(0.6518)(0.025)^2 = 1.0185 \cdot 10^{-1}\ m \cdot s^{-1}. \tag{5.60}$$

The Reynolds number of the flow for the center line velocity equals $Re_c = 4985$. Fig. 5.20 shows an estimate of the L_2 norm error for the inlet velocity field. It is calculated for five different mesh configurations in which only N_r varies (i.e. $N_r(i) = 3^i = 3, 9, 27, 81, 243$ for $i = 1, ..., 5$). The slope is equal to 2 as expected for the given discretization scheme.

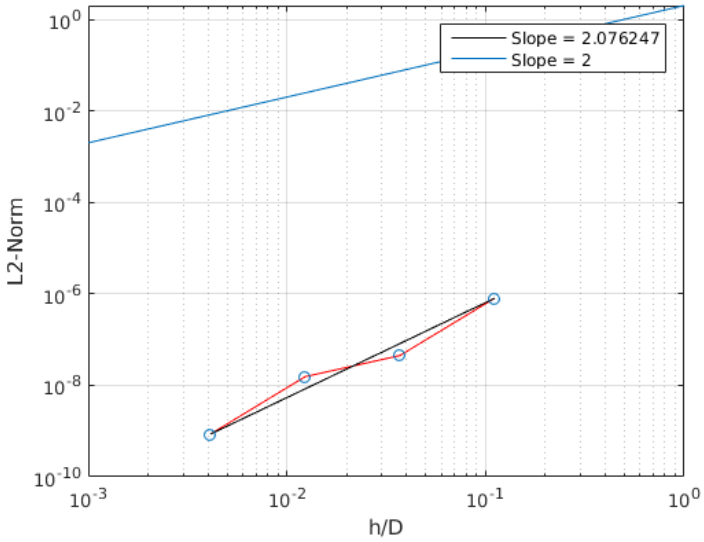


Figure 5.20: L_2 norm error for the inlet velocity field

Chapter 6

Poiseuille flow with a variable viscosity

In this chapter, analytical solutions are calculated for Poiseuille flows with a variable viscosity formulation. The exact equations for the velocity field are obtained for both Poiseuille flow in a rectangular channel (2D) and Poiseuille flow in a pipe (3D) with a variable viscosity in vertical/radial-direction. By incorporating the variable viscosity formulation in the algorithm, a step towards LES computations (for the subgrid scale model) is taken. In addition to this, it is also necessary to perform convergence studies by extending the diffusive transport term to include variable viscosity in order to be able to model two-phase flow in future.

6.1 In Cartesian coordinates

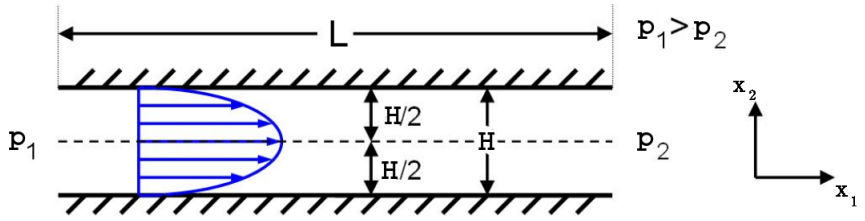


Figure 6.1: Poiseuille flow in Cartesian coordinates

Fig. 6.1 illustrates the Poiseuille flow in Cartesian coordinates. The velocity $\mathbf{u} = (u_1, u_2, u_3)$ varies only in x_2-direction and pressure drops linearly in x_1-direction. The continuity equation is given by

$$\frac{\partial u_1}{\partial x_1} = 0, \tag{6.1}$$

where velocity components in other directions and pressure are given below:

$$u_2 = u_3 = 0, \tag{6.2}$$
$$u_1 = u_1(x_2), \tag{6.3}$$
$$\frac{\partial p}{\partial x_2} = \frac{\partial p}{\partial x_3} = 0, \tag{6.4}$$
$$p = p(x). \tag{6.5}$$

The Navier-Stokes equations for a Poiseuille flow with a variable viscosity in Cartesian coordinates are given by

$$-\frac{\partial p}{\partial x_1} + \frac{\partial}{\partial x_2}\left(\mu(x_2)\frac{\partial u_1}{\partial x_2}\right) = 0, \tag{6.6}$$

$$\frac{\partial}{\partial x_2}\left(\mu(x_2)\frac{\partial u_1}{\partial x_2}\right) = \frac{\partial p}{\partial x_1}, \tag{6.7}$$

$$\mu'\frac{\partial u_1}{\partial x_2} + \mu\frac{\partial^2 u_1}{\partial x_2^2} = C, \tag{6.8}$$

where C is a constant representing the constant pressure gradient. Let us define $w = \frac{\partial u_1}{\partial x_2}$ and rewrite Eq. (6.8) by dividing both sides with μ:

$$w' + \frac{\mu'}{\mu}w = \frac{C}{\mu}, \tag{6.9}$$

In order to make linear first order ordinary differential Eq. (6.9) integrable, the equation is multiplied with an integrating factor $I(x_2)$:

$$y' + p(x_2)x_2 = q(x_2), \tag{6.10}$$
$$(I(x_2)x_2)' = I(x_2)q(x_2), \tag{6.11}$$
$$I(x_2) = e^{\int p(x_2)\ dx_2}. \tag{6.12}$$

Eq. (6.9) can be integrated to

$$w(x_2) = \frac{\partial u_1}{\partial x_2} = \frac{1}{I(x_2)}\int \frac{C}{\mu} I(x_2)\ dx_2, \tag{6.13}$$

where $I(x_2) = e^{\int \frac{\mu'}{\mu} dx_2}$. The differential equation can be integrated to:

$$u_1(x_2) = \int w(x_2)\ dx_2 = \int \frac{1}{e^{\int \frac{\mu'}{\mu}\ dx_2}}\left(\int \frac{C}{\mu} e^{\int \frac{\mu'}{\mu}\ dx_2}\ dx_2\right)\ dx_2. \tag{6.14}$$

The first derivative can be rewritten:

$$w(x_2) = \frac{\partial u_x}{\partial x_2} = \frac{1}{e^{\int \frac{\mu'}{\mu}\ dx_2}}\int \frac{C}{\mu} e^{\int \frac{\mu'}{\mu}\ dx_2}\ dx_2 = 0. \tag{6.15}$$

6.1.1 Linear change in viscosity

The dynamic viscosity can be defined by assuming a linear change (a linear distribution is chosen for simplicity) in x_2-direction:

$$\mu(x_2) = ax_2 + b, \tag{6.16}$$

where a and b are constants. Then, Eq. (6.15) can be written as follows:

$$\frac{\partial u_1}{\partial x_2} = \frac{1}{ax_2 + b} \int \frac{C}{\cancel{ax_2 + b}} \cancel{(ax_2 + b)}\, dx_2 = 0, \tag{6.17}$$

$$\frac{\partial u_1}{\partial x_2} = \frac{Cy}{ax_2 + b} + k_1 = 0, \tag{6.18}$$

where k_1 is the constant of integration. The velocity is subject to the following boundary conditions:

$$u_1(x_2) = 0 \text{ at } x_2 = H/2, \text{ (no-slip boundary condition at the wall)}, \tag{6.19}$$

$$\frac{\partial u_1}{\partial x_2} = 0 \text{ at the center of the channel } x_2 = 0, \text{ (axial symmetry)}. \tag{6.20}$$

Axial symmetry boundary condition yields:

$$k_1 = -\frac{Cx_2}{ax_2 + b}\bigg|_{x_2=0} = 0, \tag{6.21}$$

$$w(x_2) = \frac{\partial u_x}{\partial x_2} = \frac{Cx_2}{ax_2 + b}. \tag{6.22}$$

The solution for the velocity is calculated by taking the integral of Eq. (6.22):

$$u_1(x_2) = \int w(x_2)\, dx_2 = \int \frac{Cx_2}{ax_2 + b}\, dx_2 = 0. \tag{6.23}$$

Application of the no-slip boundary condition yields the following:

$$\begin{aligned} u_1(H/2) &= C\left(\frac{ax_2 - b\ log(ax_2 + b)}{a^2}\right)\bigg|_{x_2=H/2} + k_2 \\ &= C\frac{\frac{aH}{2} - b\ log(\frac{aH}{2} + b)}{a^2} + k_2 = 0. \end{aligned} \tag{6.24}$$

Therefore, the constant is

$$k_2 = -C\left(\frac{\frac{aH}{2} - b\ log(\frac{aH}{2} + b)}{a^2}\right). \tag{6.25}$$

The solution for the velocity of the fluid is

$$\begin{aligned} u_1(x_2) &= \frac{C}{a^2}\left(ax_2 - b\ log(ax_2 + b) - \left(\frac{aH}{2} - b\ log\left(\frac{aH}{2} + b\right)\right)\right) \\ &= \frac{C}{a^2}\left(a\left(x_2 - \frac{H}{2}\right) - b\ log\left(\frac{ax_2 + b}{\frac{aH}{2} + b}\right)\right). \end{aligned} \tag{6.26}$$

The maximum fluid velocity (the center line velocity) is:

$$u_{max} = \frac{c}{a^2}\left(-\frac{aH}{2} - b\, log\left(\frac{b}{\frac{aH}{2}+b}\right)\right). \tag{6.27}$$

Fig. 6.2 represents the initial analytical velocity profile for a given constant pressure gradient of $C = -9.60 \cdot 10^{-2}\ N \cdot m^{-3}$ and the desired analytical and numerical velocity profiles. The final numerical velocity profile exactly matches the analytical one for a given pressure gradient of $C = -1.50 \cdot 10^{-1}\ N \cdot m^{-3}$ and $Re_c = 2336$. The initial volumetric flow rate corresponding to this velocity profile is $Q = 1.00 \cdot 10^{-3}\ m^3 \cdot s^{-1}$. Fig. 6.3 shows the initial analytical velocity profile that is calculated for a constant

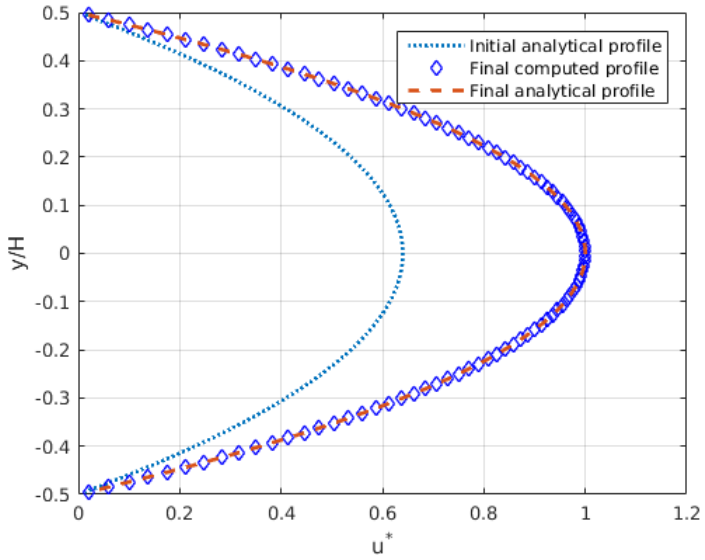


Figure 6.2: Initial analytical, final analytical and numerical dimensionless velocity profiles (with respect to the center line velocity) for a constant viscosity (i.e. a is very small) and $Re_c = 2336$

pressure gradient of $C = -9.60 \cdot 10^{-2}\ N \cdot m^{-3}$ and compares the desired analytical and numerical velocity profiles for the prescribed flow rate. The dynamic viscosity ranges from $1 \cdot 10^{-3}\ N \cdot s \cdot m^{-2}$ (at $y = 0$) to $5 \cdot 10^{-4}\ N \cdot s \cdot m^{-2}$ (at $y = H/2$). The desired pressure gradient is $C = -1.50 \cdot 10^{-1}\ N \cdot m^{-3}$ and flow rate is $Q = 2.55 \cdot 10^{-3}\ m^3 \cdot s^{-1}$. Fig. 6.4 illustrates the initial analytical velocity profile for a constant pressure gradient of $C = -9.60 \cdot 10^{-2}\ N \cdot m^{-3}$ and compares the desired analytical and numerical velocity profiles for the prescribed flow rate. In this case, the dynamic viscosity changes from center to the walls ($1 \cdot 10^{-4}\ N \cdot s \cdot m^{-3}$ at $y = 0$ and $1 \cdot 10^{-4}\ N \cdot s \cdot m^{-2}$ at $y = H/2$). The lower limit for the viscosity is smaller than the first two cases. Therefore, the analytical velocity profile results in a larger flow rate than previous cases ($Q = 6.41 \cdot 10^{-3}\ m^3 \cdot s^{-1}$).

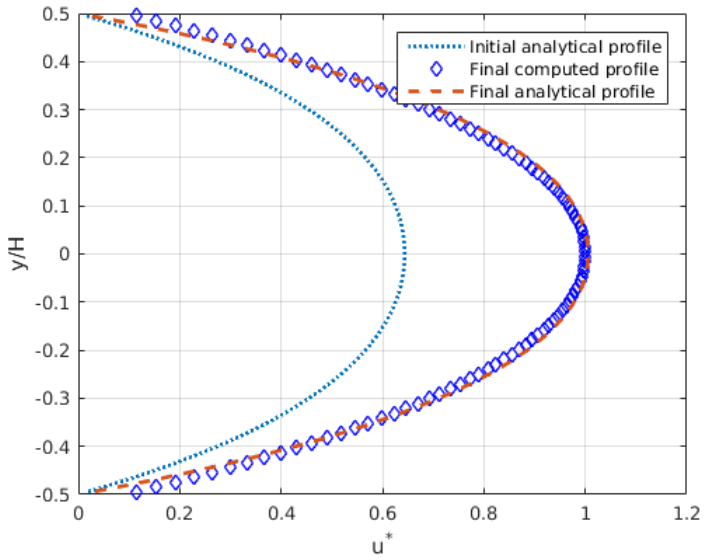


Figure 6.3: Initial analytical, final analytical and numerical dimensionless velocity profiles (with respect to the center line velocity) for $\mu = [5 \cdot 10^{-4}, 1 \cdot 10^{-3}]\ kg \cdot s^{-1} \cdot m^{-1}$

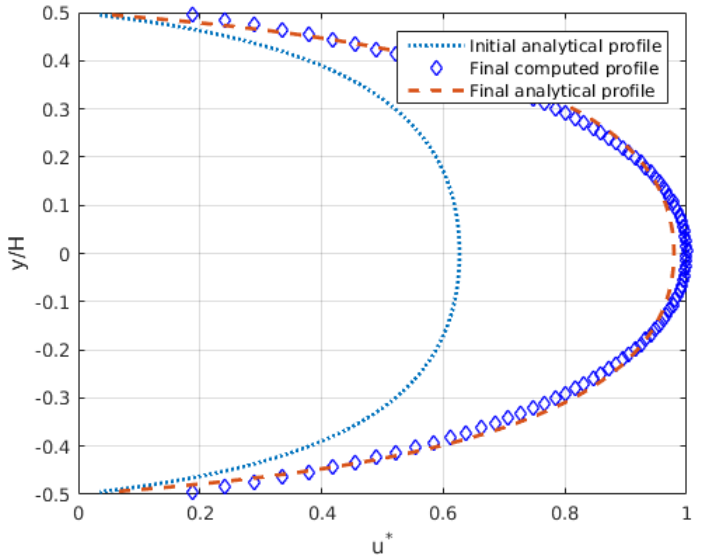


Figure 6.4: Initial analytical, final analytical and numerical dimensionless velocity profiles (with respect to the center line velocity) for $\mu = [1 \cdot 10^{-4}, 1 \cdot 10^{-3}]\ kg \cdot s^{-1} \cdot m^{-1}$

6.2 In cylindrical coordinates

Fig. 6.5 represents the Poiseuille flow in cylindrical coordinates. The aim of this test is to validate the discretization of the Navier-Stokes equations with a variable viscosity.

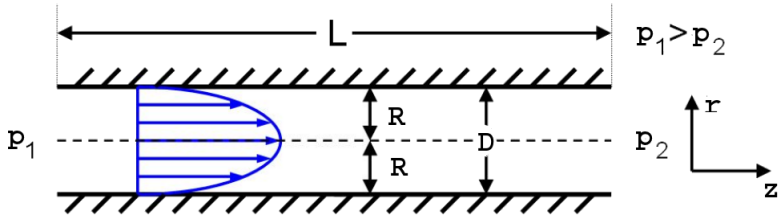


Figure 6.5: Poiseuille flow in cylindrical coordinates

The velocity of the fluid varies only in radial direction and pressure drops linearly in axial direction. The continuity equation is given by

$$\frac{\partial u_z}{\partial z} = 0, \tag{6.28}$$

(6.29)

where velocity in other directions and pressure are given below:

$$u_r = u_\theta = 0, \tag{6.30}$$
$$u_z = u_z(r), \tag{6.31}$$
$$\frac{\partial p}{\partial r} = \frac{\partial p}{\partial z} = 0, \tag{6.32}$$
$$p = p(z). \tag{6.33}$$

The Navier-Stokes equations for a Poiseuille flow with a variable viscosity in cylindrical coordinates is

$$-\frac{\partial p}{\partial z} + \frac{1}{r}\frac{\partial}{\partial r}\left(\mu(r) r \frac{\partial u_z}{\partial r}\right) = 0, \tag{6.34}$$
$$\frac{\partial}{\partial r}\left(\mu(r) r \frac{\partial u_z}{\partial r}\right) = r\frac{\partial p}{\partial z}, \tag{6.35}$$
$$\mu' r \frac{\partial u_z}{\partial r} + \mu \frac{\partial u_z}{\partial r} + \mu r \frac{\partial^2 u_z}{\partial r^2} = rC, \tag{6.36}$$

where C is a constant representing the constant pressure gradient. Substituting $w = \frac{\partial u_z}{\partial r}$ and rewriting Eq. (6.36) by dividing both sides with μr yields

$$w' + \frac{w}{r} + \frac{\mu'}{\mu} w = \frac{C}{\mu}, \tag{6.37}$$
$$w' + w\left(\frac{1}{r} + \frac{\mu'}{\mu}\right) = \frac{C}{\mu}. \tag{6.38}$$

An integrating factor $I(r)$ is defined which makes linear first order ordinary differential Eq. (6.38) integrable:

$$I(r) = e^{\int \left(\frac{1}{r} + \frac{\mu'}{\mu}\right)\, dr}. \tag{6.39}$$

Then, the first derivative of velocity is obtained:

$$w(r) = \frac{\partial u_z}{\partial r} = \frac{1}{I(r)} \int \frac{C}{\mu} I(r)\, dr. \tag{6.40}$$

The following result is obtained by inserting the value of integration factor:

$$w(r) = \frac{1}{e^{\int (\frac{1}{r}+\frac{\mu'}{\mu})\ dr}} \int \frac{C}{\mu} e^{\int (\frac{1}{r}+\frac{\mu'}{\mu})\ dr}\ dr. \tag{6.41}$$

The differential equation can be integrated to:

$$u_z(r) = \int w(r)\ dr = \int \frac{1}{e^{\int (\frac{1}{r}+\frac{\mu'}{\mu})\ dr}} \left(\int \frac{C}{\mu} e^{\int (\frac{1}{r}+\frac{\mu'}{\mu})\ dr}\ dr \right)\ dr. \tag{6.42}$$

6.2.1 Linear change in viscosity

The dynamic viscosity can be also defined by assuming a linear change (it is chosen for simplicity) in r-direction:

$$\mu(r) = ar + b. \tag{6.43}$$

The following equation for the first derivative of velocity is obtained after replacing dynamic viscosity with its definition:

$$\frac{\partial u_z}{\partial r} = \frac{1}{r(ar+b)} \int \frac{C}{\cancel{(ar+b)}} \cancel{r(ar+b)}\ dr. \tag{6.44}$$

$$\frac{\partial u_z}{\partial r} = \frac{1}{r(ar+b)} \left(\frac{Cr^2}{2} + d_1 \right), \tag{6.45}$$

where d_1 is the constant of integration. It should be noted that there exists a singularity at the center of the pipe ($r = 0$). The velocity is subject to the following boundary conditions:

$$\frac{\partial u_z}{\partial r} = 0 \text{ at the center of the channel } r = 0, \text{ (axial symmetry)}, \tag{6.46}$$

$$u_z(r) = 0 \text{ at } r = R, \text{ (no-slip boundary condition at the wall)}. \tag{6.47}$$

Axial symmetry boundary condition yields:

$$d_1 = \left. \frac{-Cr^2}{2} \right|_{r=0} = 0. \tag{6.48}$$

Therefore, the first derivative of velocity is

$$\frac{\partial u_z}{\partial r} = \frac{Cr^2}{2r(ar+b)}. \tag{6.49}$$

The differential equation can be integrated to obtain the solution for velocity:

$$u_z(r) = \int w(r)\ dr = \int \frac{\partial u_z}{\partial r}\ dr = \int \frac{Cr^2}{2r(ar+b)}\ dr. \tag{6.50}$$

The no-slip boundary condition yields the following equation:

$$\begin{aligned} u_z(R) &= \left. \frac{C\big(ar - b\ log(ar+b)\big)}{2a^2} \right|_{r=R} + d_2 \\ &= \frac{C\big(aR - b\ log(aR+b)\big)}{2a^2} + d_2 = 0. \end{aligned} \tag{6.51}$$

As a result, the constant of integration is

$$d_2 = -\frac{C\big(aR - b\ log(aR+b)\big)}{2a^2}. \tag{6.52}$$

The equation for the velocity of fluid moving through the pipe as a function of distance from the center of the pipe is

$$\begin{aligned} u_z(r) &= \frac{C\big(ar - b\ log(ar+b)\big)}{2a^2} - \frac{C\big(aR - b\ log(aR+b)\big)}{2a^2} \\ &= \frac{C}{2a^2}\Big(\big(ar - b\ log(ar+b)\big) - \big(aR - b\ log(aR+b)\big)\Big) \\ &= \frac{C}{2a^2}\Big(a(r-R) - b\ \big(log(ar+b) - log(aR+b)\big)\Big) \\ &= \frac{C}{2a^2}\Big(a(r-R) - b\ log\frac{(ar+b)}{(aR+b)}\Big). \end{aligned} \tag{6.53}$$

The exact volumetric flow rate is calculated by integrating the velocity profile Eq. (6.53):

$$\begin{aligned} Q_{exact} &= \int_0^R 2\pi \frac{C}{2a^2}\Big(a(r-R) - b\ log\frac{(ar+b)}{(aR+b)}\Big) r dr \\ &= -\frac{\pi C\Big(6b\,(ar-b)\,(ar+b)\ln\left(|ar+b|\right) - 4a^3r^3}{12a^4} \\ &\quad \frac{-3a^2\left(2b\ln\left(aR+b\right) - 2aR + b\right)r^2 + 6ab^2 r\Big)}{12a^4}\Bigg|_{r=0}^{r=R} \\ &= -\frac{\pi C\Big(6b\,(aR-b)\,(aR+b)\ln\left(|aR+b|\right) - 4a^3R^3}{12a^4} \\ &\quad \frac{-3a^2\left(2b\ln\left(aR+b\right) - 2aR + b\right)R^2 + 6ab^2 R - 6b^3\ln|b|\Big)}{12a^4}. \end{aligned} \tag{6.54}$$

The fluid velocity is maximum at the center of the pipe:

$$u_{max} = \frac{C}{2a^2}\Big(-aR - b\ log\frac{b}{(aR+b)}\Big). \tag{6.55}$$

Fig. 6.6 represents the initial analytical velocity profile for a given constant pressure gradient of $C = -6.40 \cdot 10^{-1}\ N \cdot m^{-3}$ (the initial volumetric flow rate corresponding to this velocity profile is $Q = 9.81 \cdot 10^{-5}\ m^3 \cdot s^{-1}$) and the desired analytical and numerical velocity profiles for the Reynolds number based on center line velocity $Re_c = 997$. Fig. 6.7 compares the initial analytical velocity profile and the desired analytical and numerical velocity profiles for a given pressure gradient of $C = -8.00 \cdot 10^{-1}\ N \cdot m^{-3}$ and a flow rate of $Q = 2.07 \cdot 10^{-4}\ m^3 \cdot s^{-1}$ (with the same Reynolds number $Re_c = 997$). The flow rate of the initial analytical profile in Fig. 6.8 is $Q = 4.78 \cdot 10^{-5}\ m^3 \cdot s^{-1}$. The desired analytical and computed velocity profiles are compared in Fig. 6.8 for a given pressure gradient of $C = -4.00 \cdot 10^{-1}\ N \cdot m^{-3}$ and flow rate of $Q = 2.82 \cdot 10^{-4}\ m^3 \cdot s^{-1}$ (with the same

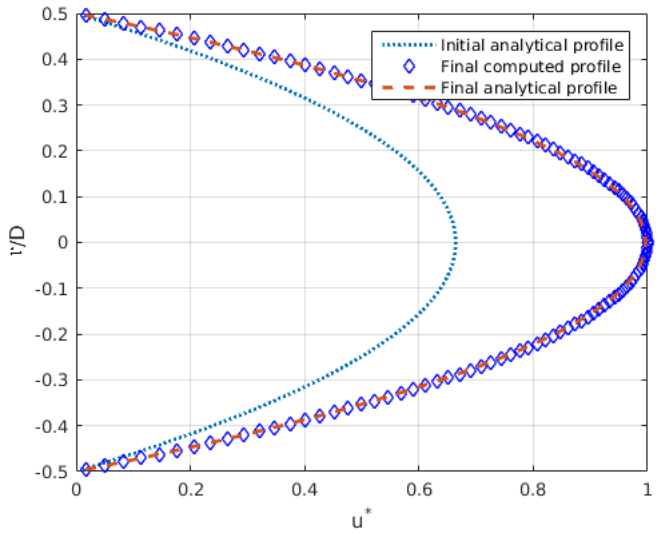


Figure 6.6: Initial analytical, final analytical and numerical dimensionless velocity profiles (with respect to the center line velocity) for a constant viscosity (i.e. a is very small)

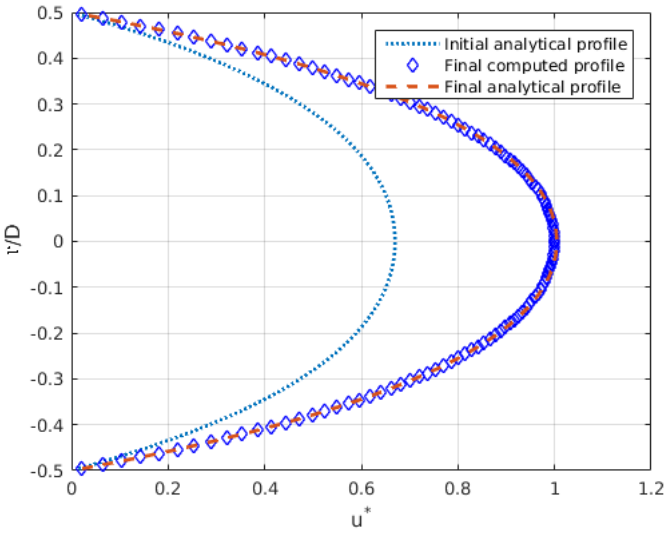


Figure 6.7: Initial analytical, final analytical and numerical dimensionless velocity profiles (with respect to the center line velocity) for $\mu = [5 \cdot 10^{-4}, 1 \cdot 10^{-3}]\ kg \cdot s^{-1} \cdot m^{-1}$

Reynolds number $Re_c = 997$). Fig. 6.9 shows velocity profiles with variable viscosity and with constant viscosity. A comparison of Poiseuille profile with the same flow rate as a profile with linearly assumed viscosity is realized in order to identify that the velocity profile with linearly assumed viscosity results in a non-parabolic profile. Fig. 6.10 shows an estimate for the L_2 norm error of the inlet velocity field. The Reynolds number of the final flow for the center-line velocity is $Re_c = 255$. The error is calculated for five

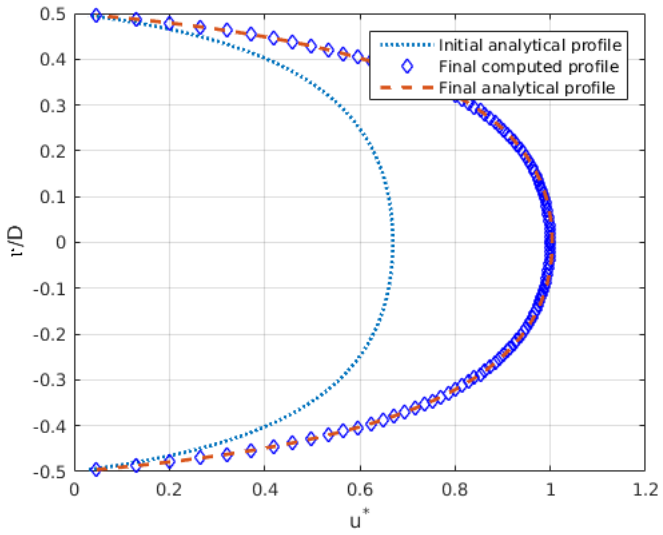


Figure 6.8: Initial analytical, final analytical and numerical dimensionless velocity profiles (with respect to the center line velocity) for $\mu = [1 \cdot 10^{-4}, 1 \cdot 10^{-3}]\ kg \cdot s^{-1} \cdot m^{-1}$

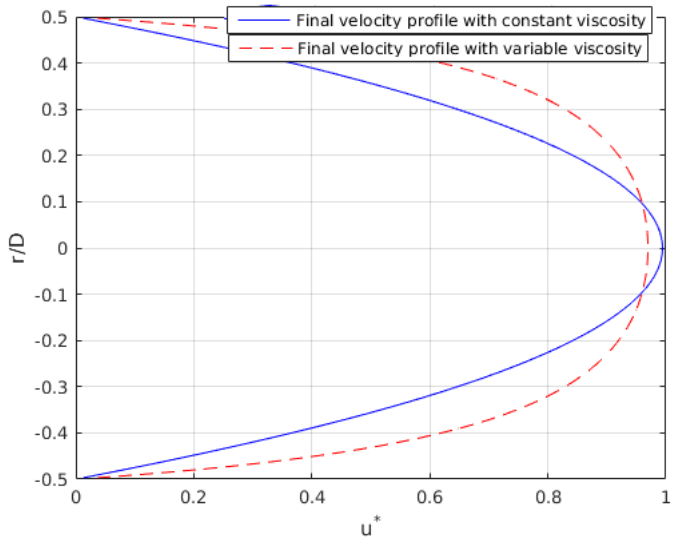


Figure 6.9: Final numerical dimensionless velocity profiles (normalized with respect to the center line velocity) with linearly assumed viscosity $\mu = [1 \cdot 10^{-4}, 1 \cdot 10^{-3}]\ kg \cdot s^{-1} \cdot m^{-1}$ and with constant viscosity $\mu_{cnst} = 1 \cdot 10^{-3}\ kg \cdot s^{-1} \cdot m^{-1}$

different mesh configurations in which only N_r varies (i.e. $N_r(i) = 3^i = 3, 9, 27, 81, 243$ for $i = 1, ..., 5$).

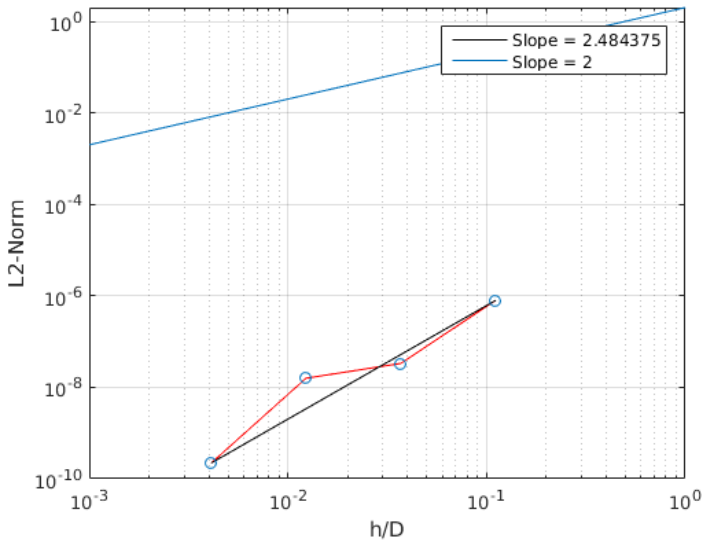


Figure 6.10: L_2 norm error for the inlet velocity field with a variable viscosity formulation

6.2.2 Choice of input parameters

There are mainly two possible input parameters for the numerical solution. The constant pressure gradient and volumetric flow rate are compared to identify which of these results in a faster convergence to the exact solution.

The numerical solution shown in Fig. 6.11 converges to the exact solution in 161 s for a given tolerance and flow rate of $Q_{final} = 1.5 \cdot 10^{-5}\ m^3 \cdot s^{-1}$. The initial flow rate is chosen to be $Q_{initial} = 1.0 \cdot 10^{-5}\ m^3 \cdot s^{-1}$ and the initial value for the tolerance $|Q - \mathbf{Ru^*}|/Q = 0.33363$.

The numerical solution shown in Fig. 6.12 converges to the exact solution in 193.2 s for a given tolerance and flow rate of $Q_{final} = 1.5 \cdot 10^{-5}\ m^3 \cdot s^{-1}$. The initial flow rate is chosen to be $Q_{initial} = 1.0 \cdot 10^{-5}\ m^3 \cdot s^{-1}$ and the initial value for the tolerance is $|Q - \mathbf{Ru^*}|/Q = 0.498112$.

When a constant pressure gradient of $C_{final} = -2.125 \cdot 10^{-2}\ N \cdot m^{-3}$ is prescribed as an input, the total time needed to converge to the exact solution is 193.2 s as shown in Fig. 6.12. The initial pressure gradient is chosen to be $C_{initial} = -1.417 \cdot 10^{-1}\ N \cdot m^{-3}$ and the initial value for the tolerance is $tol_{init} = |Q - \mathbf{Ru^*}|/Q = 0.498112$. The initial tolerance value is defined such that it will be same for both cases.

It is shown that choice of input parameter (a pressure gradient or flow rate) has no influence on the rate of convergence. The total time of the simulation is same for both of the cases.

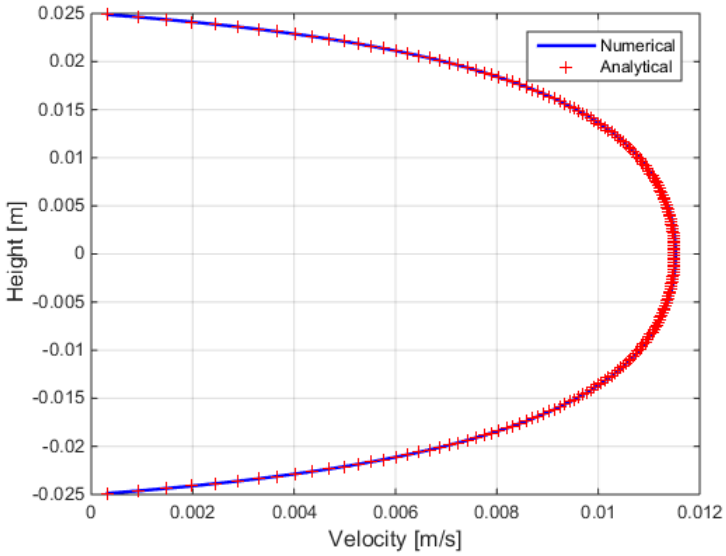


Figure 6.11: Comparison of the desired analytical profile and calculated velocity profile at iteration 1932 for the desired flow rate $Q_{final} = 1.5 \cdot 10^{-5}\ m^3 \cdot s^{-1}$

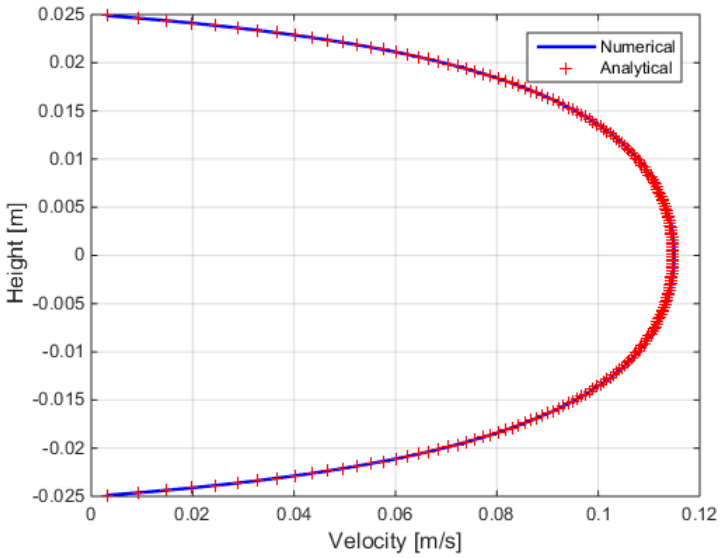


Figure 6.12: Comparison of the desired analytical profile and calculated velocity profile at iteration 1932 for the desired flow rate $C_{final} = -2.125 \cdot 10^{-2}\ N \cdot m^{-3}$

6.3 A non-symmetric pressure matrix

Figures 6.13, 6.14 and 6.15 visualize the corresponding two dimensional matrices of sizes 60×60. Fig. 6.13 represents the symmetric Poisson matrix $\boldsymbol{DM^{-1}G}$. A new term

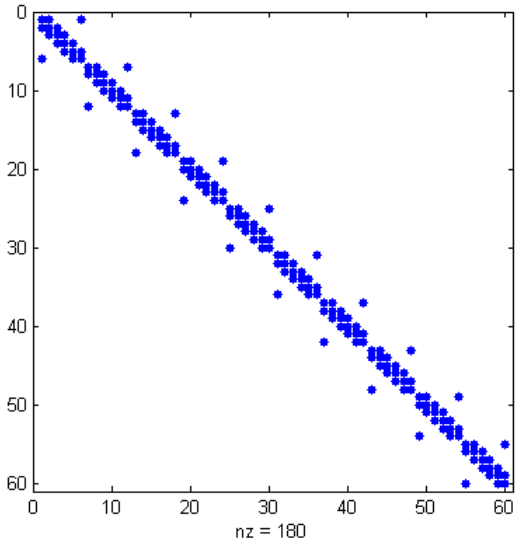


Figure 6.13: Visualization of the laplacian matrix

$\boldsymbol{DM^{-1}}\left(\frac{\bar{\boldsymbol{G}}\boldsymbol{RM^{-1}G}}{\boldsymbol{RM^{-1}}\bar{\boldsymbol{G}}}\right)$, which is a result of the periodic boundary condition, is visualized in Fig. 6.14. The fill pattern seems symmetric, but the coefficients are not symmetric because of the $\boldsymbol{R}$ operator. This new term is subtracted from the original Poisson equation left hand side matrix $\boldsymbol{DM^{-1}}\left(\boldsymbol{G}-\boxed{\frac{\bar{\boldsymbol{G}}\boldsymbol{RM^{-1}G}}{\boldsymbol{RM^{-1}}\bar{\boldsymbol{G}}}}\right)$. Fig. 6.15 shows the effect of the additional part, which is coming from the periodic boundary conditions, on the left hand side of the Poisson equation. This resulting matrix is also not symmetric due to the $\boldsymbol{R}$ operator.

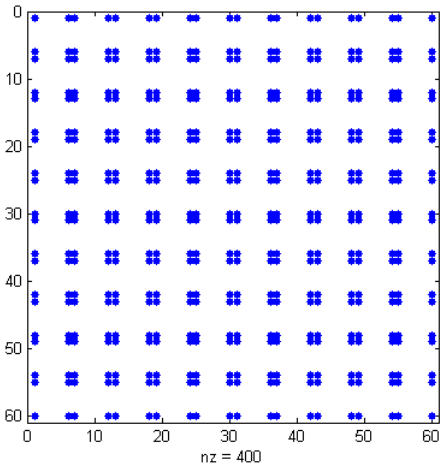


Figure 6.14: Visualization of the additional sparse matrix coming from periodic boundary conditions

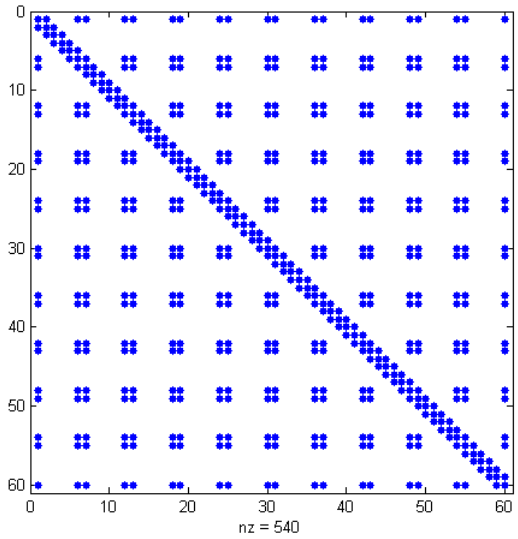


Figure 6.15: Visualization of the augmented sparse matrix

6.4 Preconditioners for GMRES solvers

The momentum equatıon is preconditioned by scaling the left and right-hand side matrices with the same constant [1]. This procedure increases the efficiency of the solver for larger time steps while using the generalized minimal residual method (GMRES). The pressure matrix in Eq. (5.33) is non-symmetric. Therefore, the GMRES method is also used to solve the modified Laplactan-type equation for pressure. However, from the local convergence behaviour of GMRES (see Fig. 6.16) it appears that the residual remains the same in the first and last 30 iterations [2]. For this reason, GMRES is preconditioned in order to increase the convergence rate. The preconditioning transforms the original linear system into another linear system, which has the same solution, to make the iterative solver cheaper and faster.

An incomplete LU decomposition of the matrix $\boldsymbol{DM^{-1}G}$ is used as a preconditioner (this approach was suggested in the report of Segal et al. (1994) [25]). Fig. 6.16 indicates that it is a rather good preconditioner. The size of the pressure matrix used in Fig. 6.16 is 68 $\times$ 48. The relative residuals of the GMRES algorithm show two stagnation periods followed by a rapid convergence at the end of these stagnation periods as shown in Fig. 6.16. The PGMRES converges to a solution in 4 iterations, whereas the GMRES algorithm without preconditioning converges in 68 iterations. The PGMRES method is considerably cheaper and faster than the GMRES method without preconditioning. With the same amount of unknowns, 3264, the average number of iterations for the pressure equations decreased

[1] The preconditioning is done by scaling the diagonal entries to unity.
[2] The residual of a finer numerical resolution in three dimensions actually remains the same for indefinite number of iterations.

from 208 to 98 in 3D `Fortran` code. This demonstrates that the incomplete LU of the symmetric part of whole pressure matrix is a rather good preconditioner for the complete system.

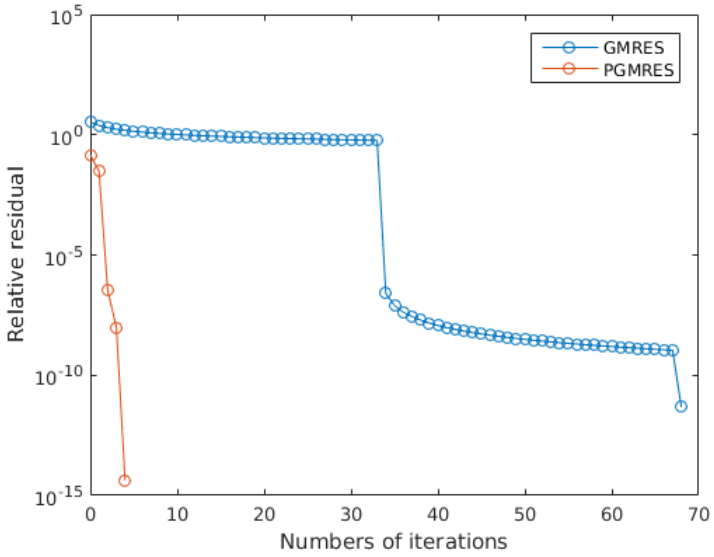
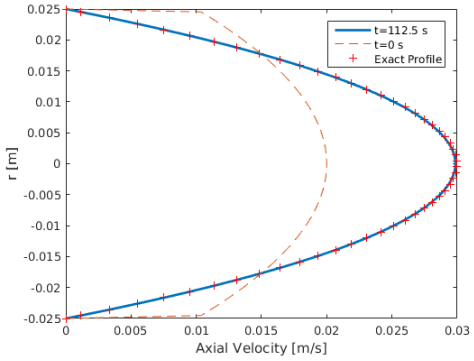


Figure 6.16: Comparison of relative residuals of preconditioned `GMRES` (`PGMRES`) and `GMRES` algorithms without a preconditioner

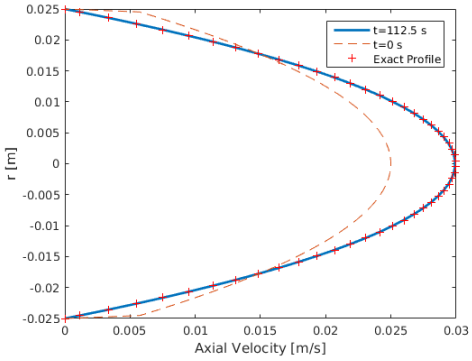
6.5 Velocity profiles

The prescribed flow rate is defined such that the Reynolds number based on the center-line velocity equals $Re_c = 1495.5$. Two different initial flow rates are chosen for both constant viscosity and for linearly varying viscosity in radial direction between $1 \cdot 10^{-3}$ and $1 \cdot 10^{-4}$ to compare the rate of convergence of two different initial velocity profiles in these cases. For the first case and the second case, the initial flow rate and the Reynolds number are $Q_{initially} = 9.82 \cdot 10^{-6}\ m^3 \cdot s^{-1}$, $Re_c = 498.5$ and $Q_{initially} = 1.96 \cdot 10^{-5}\ m^3 \cdot s^{-1}$, $Re_c = 997$ respectively. The final flow rates for the Poiseuille flow with a constant viscosity and variable viscosity are $Q_{final} = 2.94 \cdot 10^{-5}\ m^3 \cdot s^{-1}$ and $Q_{final} = 3.91 \cdot 10^{-5}\ m^3 \cdot s^{-1}$ respectively.

The first test case is simulated with a constant viscosity assumption. Both initial profiles converged to the desired velocity profile at different convergence rates as shown in Figures 6.17a and 6.17b. Figures 6.18a and 6.18b represents the converged velocity profiles for the second test case, which is formulated with a variable viscosity.

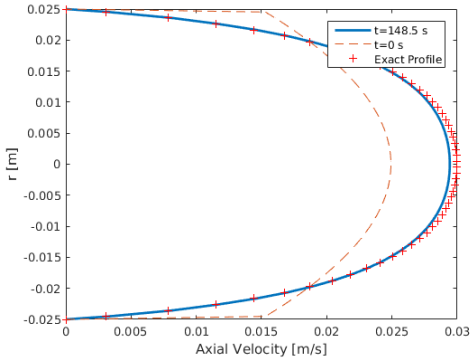


(a) Initial and final velocity profiles for $Re_c = 498.5$

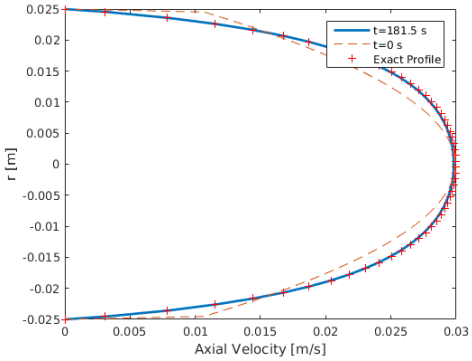


(b) Initial and final velocity profiles for $Re_c = 997$

Figure 6.17: Fully-developed laminar velocity profiles at first and last time steps for a constant viscosity for $Re_c = 498.5$ and $Re_c = 997$



(a) Initial and final velocity profiles for $Re_c = 498.5$



(b) Initial and final velocity profiles for $Re_c = 997$

Figure 6.18: Fully-developed laminar velocity profiles at first and last time steps for a variable viscosity for $Re_c = 498.5$ and $Re_c = 997$

Chapter 7

Numerical simulation of turbulent flow in pipes

In this chapter, results of DNS and LES computations for turbulent pipe flows are presented. The aim of these numerical simulations is to assess the current discretization for fully developed turbulent pipe flow. In order to demonstrate that the developed numerical method is able to accurately describe fully developed turbulent pipe flow, computed results provided by Eggels [1] are compared with the results of this study. The main intention of this chapter is to replicate the results of Eggels. Therefore, the test cases are actually chosen according to the cases presented in [1]. The choice for grid resolution and time step sizes are chosen to make a comparison possible with the results of Eggels. However, parameters of some cases deviate from the parameters chosen by Eggels because of the differences between the current algorithm and the one from Eggels. The continuity and momentum equations are solved in a cylindrical pipe with a diameter D and length L for both numerical simulations. The pipe length is not chosen the same in both DNS and LES. The length of the pipe in DNS is shorter than LES due to limitations imposed by the finer numerical resolution. The present numerical results illustrate that they are in agreement with the results of [1]. These numerical simulations constitute the first steps towards the simulation of turbulent two-phase flow.

7.1 Direct Numerical Simulation of fully developed pipe flow

7.1.1 Introduction to DNS computations

The turbulent pipe flow is simulated in a cylindrical geometry with a diameter D and a length $L = D$. The Reynolds numbers based on the mean velocity u_b, the Reynolds numbers based on the wall shear stress velocity u^*, and the Reynolds number based on the center line velocity u_c are $Re_b = 5300$, $Re_* = 360$ and $Re_c = 6950$ respectively.

The necessary periodic boundary conditions are applied for both velocity and pressure in axial and angular directions and a no-slip boundary condition is imposed at the pipe wall. The periodic boundary condition in the axial direction is imposed in order to decrease the computational requirements. These conditions will only lead to a physically realistic solution when the domain length is larger than the largest turbulent scales (i.e., the pipe length needs to be long enough to capture the largest scale structures) and if the flow is homogeneous in the axial direction. As discussed in Section 2.3.1, the required pipe length is assumed to be $L = 2.5D$ [1]. However, it is not quite practical to solve the problem while keeping $L = 2.5D$ since a longer pipe length results in large number of unknowns (i.e., for $L = 5D$ the total number of grid cell is $96 \times 128 \times 256$ based on the computations of Eggels [1]). Therefore, the periodic pipe length is reduced to $L = D$ such that the available computational resources and the inclusion of even the largest scale structures do not yield a problem. This reduction in the pipe length simultaneously decreases the required number of grid points in the axial direction. The two-point correlation coefficients of the velocity fluctuations are not exactly zero at $L = D$, but this procedure is inevitable due to the restriction imposed by the serial code [2].

The pressure drop over the length of the pipe, which balances the viscous friction at the wall (the time-averaged wall shear stress $\langle \tau_{rz} \rangle$), is calculated in every iteration as part of the solution. The velocity profile and the flow rate are then computed according to the pressure drop along the length of the pipe. The relation between the pressure drop and the time-averaged wall shear stress is as follows:

$$-\frac{1}{\rho}\frac{\Delta P}{\Delta Z} = \frac{4\langle \tau_{rz}(\frac{1}{2}D)\rangle}{\rho D}. \tag{7.1}$$

The mean pressure gradient -$(\nabla P)_z$ can be expressed by substituting $\langle \tau_{rz}(\frac{1}{2}D)\rangle = \rho u_*^2$:

$$-(\nabla P)_z = -\frac{1}{\rho}\frac{\Delta P}{\Delta Z} = \frac{4u_*^2}{D}. \tag{7.2}$$

The computations are initialized by describing initial conditions in order to obtain physically meaningful data fields representing the turbulent profile. This can be done by defining an appropriate initial flow rate and a large time step ($0.1t^*$) to obtain suitable data fields for the computations with the same numerical resolution. The temporal and spatial resolutions of the numerical simulations must be small enough to capture the smallest scales without any turbulence model.

According to Kolmogorov's first similarity hypothesis for very high Re, two-point statistics have a universal form and they depend on two parameters; the kinematic viscosity ν and the rate of energy dissipation ϵ. From these two quantities, characteristic length and time scales of the motion in the equilibrium range can be defined by

$$\eta = \left(\frac{\nu^3}{\epsilon}\right)^{1/4}, \tag{7.3}$$

$$\tau_\eta = \left(\frac{\nu}{\epsilon}\right)^{1/2}. \tag{7.4}$$

[1] The optimum pipe length computed by the correlation of the velocity fluctuations in terms of a one-dimensional correlation function is equal to two and a half times the diameter.

[2] The available computational resources and the serial code in the Scientific Computing group of the Delft Institute for Applied Mathematics are not adequate for a longer pipe length.

Here, the rate of energy dissipation is proportional to the kinetic energy per unit mass in the integral scale u^2 divided by the integral time scale l/u:

$$\epsilon \propto \frac{u^3}{l}. \tag{7.5}$$

The time step Δt used in DNS equals approximately $0.002t^*$ with t^* being the dimensionless time scale defined as the ratio of the pipe diameter and the wall shear stress velocity u_*. The imposed time step Δt is one order of magnitude larger than the one used in [1] since the time discretization method used in this thesis is second order accurate and does not require an additional stability criteria because of the use of a fully implicit time integration method that relaxes constraints on the time step size. The Kolmogorov time scale can be expressed in terms of t^*; $\tau_\eta \approx 0.0068t^*$. The resolution in time is sufficient enough to resolve all scales of motion since the time step Δt is smaller than both the Kolmogorov and the convective time scale $\Delta t \sim \nu/u^2 = 0.0028t^*$.

7.1.2 Results of DNS computations

The numerical resolution of case 1 is 58 × 60 × 58 grid points in radial, angular and axial directions respectively. Case 2 has 78 × 80 × 78 grid points and case 3 has 90 × 100 × 90 grid points as shown in Table 7.1. The results of the study of Eggels [1] for 96 × 128 × 256 grid points are $\langle u_z\rangle_c/u_*$=19.31, $\langle u_z\rangle_b/u_*$=14.73 and $\langle u_z\rangle_c/\langle u_z\rangle_b$=1.31 respectively. The case with the highest resolution (case 3) corresponds to Eggels but for the length of the pipe and the number of grid points in the axial direction. The mean flow quantities

case	N_r	N_θ	N_z	$\langle u_z\rangle_c/u_*$	$\langle u_z\rangle_b/u_*$	$\langle u_z\rangle_c/\langle u_z\rangle_b$
1	58	60	58	24.06	19.31	1.63
2	78	80	78	24.07	14.72	1.63
3	90	128	50	19.30	14.72	1.31

Table 7.1: Mean flow quantities obtained by DNS for the given Reynolds numbers $Re_c = \langle u_z\rangle_c D/\nu = 6950$ ($\langle u_z\rangle_c$ represents the center line velocity), $Re_b = \langle u_z\rangle_b D/\nu = 5300$ ($\langle u_z\rangle_b$ represents the mean or bulk velocity and it is obtained by numerical integration of the mean axial velocity profile using the midpoint rule), $Re_* = u_* D/\nu = 360$ (u_* is the wall shear stress velocity)

are obtained by averaging all data fields in the axial (z) and azimuthal (θ) directions and also by temporal (ensemble) averaging [3]. The bulk velocity $\langle u_z\rangle_b$ is determined as:

$$\langle u_z\rangle_b = \frac{2}{R^2}\int_0^R r\langle u_z\rangle_r \, dr, \tag{7.6}$$

with R being the pipe radius. The subscript $_r$ indicates the dependence of $\langle u_z\rangle$ on r.

Fig. 7.1 illustrates the initial normalized velocity profile and the normalized mean velocity profile with respect to the center line velocity for case 1 to demonstrate the difference between initial and final profiles.

[3] Mean flow quantities are denoted by $\langle\ \rangle$. Aberrations from mean values $\langle u\rangle$ are represented by a double prime $\langle u''\rangle$.

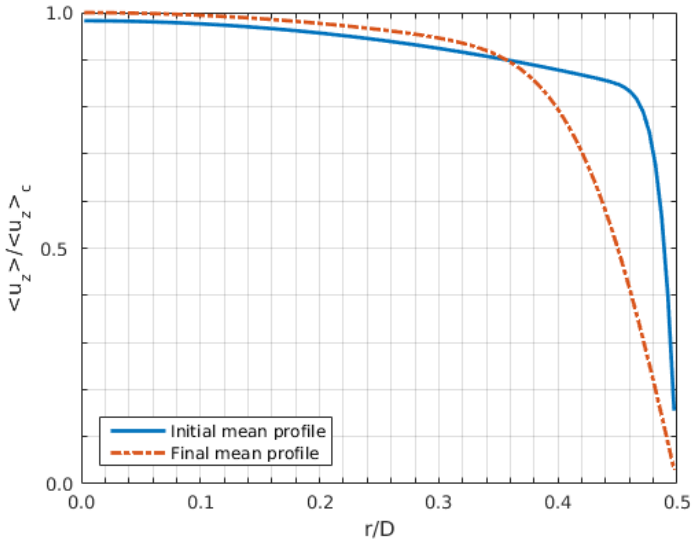


Figure 7.1: Initial normalized velocity profile and mean axial velocity normalized on the center line velocity $\langle u_z \rangle_c$ as a function of the distance from the center line for DNS-case 1 ($58 \times 60 \times 58$)

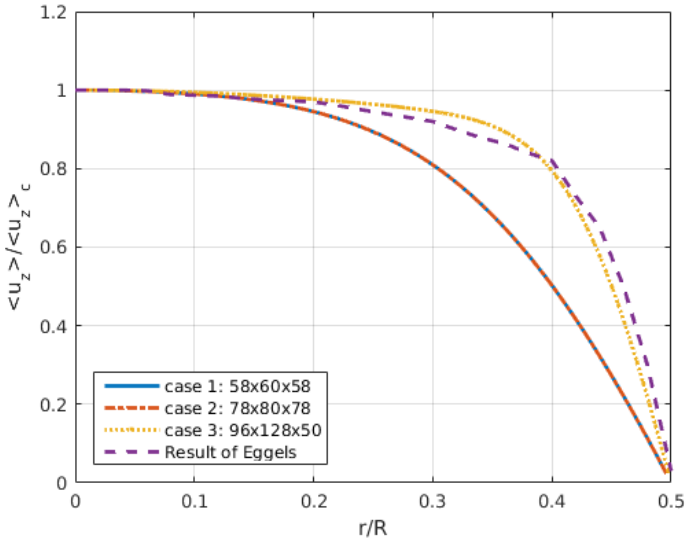


Figure 7.2: Mean axial velocity normalized on the center line velocity $\langle u_z \rangle_c$ as a function of the distance from the center line for $Re_c = 6950$

The normalized mean axial velocity profiles for all cases are shown in Fig. 7.2. The numerical results of case 3 are in good agreement with the results of [1] for all r/R. Though the numerical resolutions used in this thesis (the number of grid points in the axial direction) are smaller than the one used in [1], the results are nearly identical to the results of Eggels [1]. The results of case 1 and 2 do not accurately describe full developed

turbulent profile since the number of grid points in radial and circumferential directions for these cases are significantly lower than the required amount $90 \times 128 \times 256$ for $L = 5D$. Fig. 7.3 presents the normalized mean velocity profiles at different time locations for case

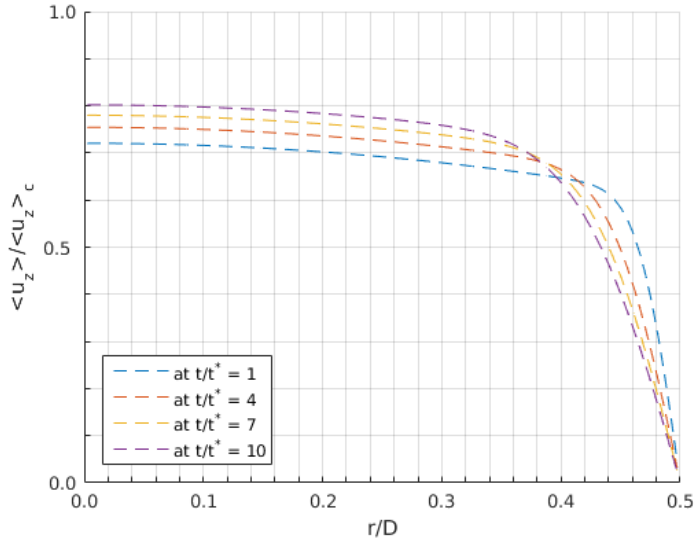


Figure 7.3: Development of mean velocity profiles nondimensionalized by the center line velocity $\langle u_z \rangle_c$ as a function of the distance from the center line through time for DNS-case 3

3. The turbulent fluctuations start to decline after $t/t^* = 7$ and get close to zero around $t/t^* = 10$.

In Fig. 7.4, the mean axial velocity nondimensionalized by the friction velocity as a function of the dimensional wall unit y^+ is shown. The velocity changes linearly in the viscous sublayer ($\langle u_z \rangle^+ = y^+$ for $y^+ < 5$) and this region is resolved in the numerical simulations. The velocity distribution changes from linear to logarithmic as the distance from the wall increases ($y^+ > 30$) and this logarithmic velocity distribution is defined with the von Kármán constant $\kappa = 0.4$ and $C^+ = 5.5$ (which is usually equal to 5.0, but for low Reynolds number flows 5.5 is found to be more suitable [1]). At this low Re, the intermediate region is not observed and the logarithmic velocity distribution is not realized in the computations. Although, the pipe length is relatively small (compared to the one used in [1] $L = 5D$), the DNS computations are still accurate for case 3 with $L = D$. In addition to this, DNS profiles of case 1 and 2 exhibit minor deviations near the wall. The reason for this discrepancy can be attributed to the length of the pipe which is smaller than the one used in [1] $L = 5D$ [4]. Other than this, another reason for the aberration in the near-wall region may be the difference in numerical resolutions of three cases (i.e., case 3 is the finest and case 1 is the coarsest). As a result, the velocity profile is better resolved in case 3 compared to case 1 and 2 in the near-wall region due to the fact that the numerical resolution in this case is sufficiently high to resolve all relevant scales of turbulent motion.

[4]The ideal pipe length computed by the correlation of the velocity fluctuations in terms of an one-dimensional correlation function is $L = 2.5D$.

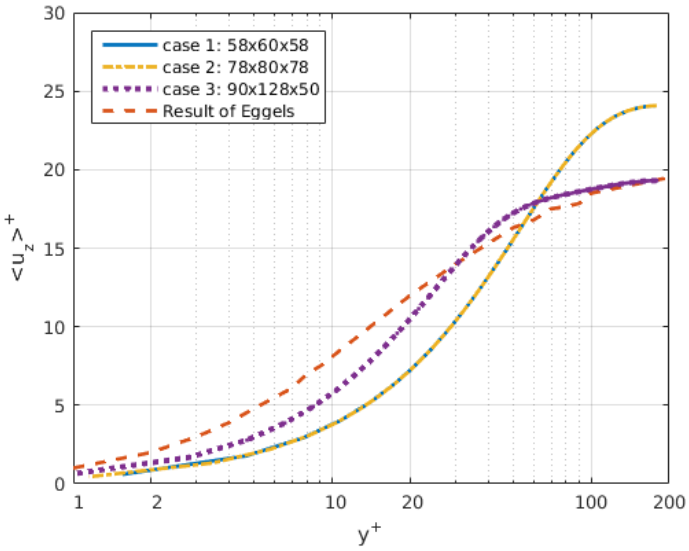


Figure 7.4: Mean axial velocity normalized with the friction velocity u_τ versus the distance from the wall

7.2 Application of LES turbulence model for fully developed pipe flow

7.2.1 Introduction to LES computations

The underlying assumptions of LES turbulence modeling have been discussed in section 2.2. An additional stress tensor term and the eddy viscosity term are introduced (refer to Eq. (2.33)). The Smargorinsky model is used to model this anisotropic residual stress tensor. The model for the eddy viscosity (Eq. (2.34)) can be rewritten in terms of the mixing length of the subgrid scale (SGS) motions l_S (also called the Smagorinsky length scale) and the deformation rate tensor as follows:

$$\nu_t := l_S^2 \sqrt{2\bar{S}_{ij}\bar{S}_{ij}} = (C_S\Delta)^2\sqrt{\bar{S}^2}, \tag{7.7}$$

where C_S is the Smagorinsky constant, the characteristic grid spacing $\Delta = (r\Delta r\Delta\theta\Delta z)^{\frac{1}{3}}$ is determined from the grid spacings Δr, $r\Delta\theta$ and Δz in radial, tangential and axial directions respectively. The deformation rate tensor of the resolved velocity field is given by:

$$\overline{S}_{ij} := \frac{1}{2}\left(\frac{\partial\overline{u}_i}{\partial x_j} + \frac{\partial\overline{u}_j}{\partial x_i}\right). \tag{7.8}$$

In cylindrical coordinates the rate of strain tensor is as follows:

$$\overline{S}_{ij} = \frac{1}{2}\begin{pmatrix} 2\dfrac{\partial \bar{u}_r}{\partial r} & \dfrac{\partial \bar{u}_\theta}{\partial r} + \dfrac{1}{r}\dfrac{\partial \bar{u}_r}{\partial \theta} - \dfrac{\bar{u}_\theta}{r} & \dfrac{\partial \bar{u}_r}{\partial z} + \dfrac{\partial \bar{u}_z}{\partial r} \\ \dfrac{\partial \bar{u}_\theta}{\partial r} + \dfrac{1}{r}\dfrac{\partial \bar{u}_r}{\partial \theta} - \dfrac{\bar{u}_\theta}{r} & 2\left(\dfrac{1}{r}\dfrac{\partial \bar{u}_\theta}{\partial \theta} + \dfrac{\bar{u}_r}{r}\right) & \dfrac{1}{r}\dfrac{\partial \bar{u}_z}{\partial \theta} + \dfrac{\partial \bar{u}_\theta}{\partial z} \\ \dfrac{\partial \bar{u}_r}{\partial z} + \dfrac{\partial \bar{u}_z}{\partial r} & \dfrac{1}{r}\dfrac{\partial \bar{u}_z}{\partial \theta} + \dfrac{\partial \bar{u}_\theta}{\partial z} & 2\dfrac{\partial \bar{u}_z}{\partial z} \end{pmatrix}.$$

The deformation of the resolved velocity field

$$\bar{S}^2 = \sum_{ij} \bar{S}_{ij}\bar{S}_{ij} \tag{7.9}$$

is derived from the given tensor inner product:

$$\bar{S}^2 = \frac{1}{4}\left(\nabla \bar{\boldsymbol{u}} + (\nabla \bar{\boldsymbol{u}})^T\right) : \left(\nabla \bar{\boldsymbol{u}} + (\nabla \bar{\boldsymbol{u}})^T\right). \tag{7.10}$$

The entries of $\bar{S}^2$ can be discretized with second order accuracy by using the central difference scheme. $\dfrac{\bar{u}_\theta}{r}$ and $\dfrac{\bar{u}_r}{r}$ terms in the radial and angular equation are approximated numerically by taking the average of two neighbor cell values. The derivatives of the radial velocity components (i.e., the contributions to the coefficients of the stress tensor) are discretized as follows:

$$\left(\frac{\partial \bar{u}_r}{\partial r}\right)\Bigg|_{i,j,k} = \frac{(\bar{u}_r)_{i+1,j,k} - (\bar{u}_r)_{i-1,j,k}}{2\Delta r}, \tag{7.11}$$

$$\left(\frac{1}{r}\frac{\partial \bar{u}_r}{\partial \theta}\right)\Bigg|_{i,j,k} = \frac{1}{r_i}\frac{(\bar{u}_r)_{i,j+1,k} - (\bar{u}_r)_{i,j-1,k}}{2\Delta \theta}, \tag{7.12}$$

$$\left(\frac{\partial \bar{u}_r}{\partial z}\right)\Bigg|_{i,j,k} = \frac{(\bar{u}_r)_{i,j,k+1} - (\bar{u}_r)_{i,j,k-1}}{2\Delta z}. \tag{7.13}$$

The derivatives of the azimuthal velocity components are calculated with the central scheme:

$$\left(\frac{\partial \bar{u}_\theta}{\partial r}\right)\Bigg|_{i,j,k} = \frac{(\bar{u}_\theta)_{i+1,j,k} - (\bar{u}_\theta)_{i-1,j,k}}{2\Delta r}, \tag{7.14}$$

$$\left(\frac{1}{r}\frac{\partial \bar{u}_\theta}{\partial \theta}\right)\Bigg|_{i,j,k} = \frac{1}{r_i}\frac{(\bar{u}_\theta)_{i,j+1,k} - (\bar{u}_\theta)_{i,j-1,k}}{2\Delta \theta}, \tag{7.15}$$

$$\left(\frac{\partial \bar{u}_\theta}{\partial z}\right)\Bigg|_{i,j,k} = \frac{(\bar{u}_\theta)_{i,j,k+1} - (\bar{u}_\theta)_{i,j,k-1}}{2\Delta z}. \tag{7.16}$$

The same procedure is applied to the derivatives of the axial components and the discretization is given by:

$$\left(\frac{\partial \bar{u}_z}{\partial r}\right)\Bigg|_{i,j,k} = \frac{(\bar{u}_z)_{i+1,j,k} - (\bar{u}_z)_{i-1,j,k}}{2\Delta r}, \tag{7.17}$$

$$\left(\frac{1}{r}\frac{\partial \bar{u}_z}{\partial \theta}\right)\Bigg|_{i,j,k} = \frac{1}{r_i}\frac{(\bar{u}_z)_{i,j+1,k} - (\bar{u}_z)_{i,j-1,k}}{2\Delta \theta}, \tag{7.18}$$

$$\left(\frac{\partial \bar{u}_z}{\partial z}\right)\Bigg|_{i,j,k+1/2} = \frac{(\bar{u}_z)_{i,j,k+1} - (\bar{u}_z)_{i,j,k}}{2\Delta z}. \tag{7.19}$$

Fig. 7.5 shows an estimate for the L_2 norm error of $\bar{S}^2$. This procedure is done just to check the accuracy and validity of $\bar{S}^2$ [5] before proceeding with computations. A parabolic velocity field $\bar{\boldsymbol{u}} = \{(4r^2+3r+2+\theta^2+z^2), (5\theta^2+4\theta+3+r^2+z^2), (6z^2+5z+4+r^2+\theta^2)\}$ is defined to verify the deformation of the resolved velocity field $\bar{S}^2$. The error is calculated for three different mesh configurations in which the number of grid points in all direction varies. The number of sample points are kept fixed and the solution on grids where the mesh width is refined with a factor 3 is compared. This means that the same points are present in the finer meshes, thus, the solution at those points can be compared with the solution of the coarsest grids. The modified filtered momentum equation can be rewritten

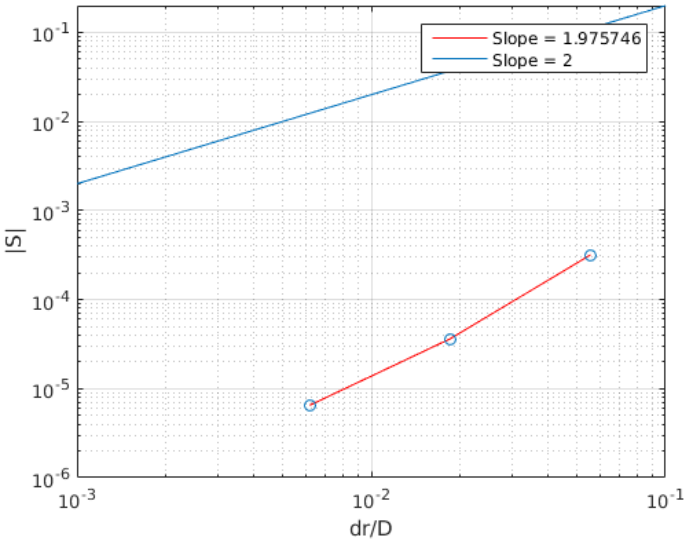


Figure 7.5: L_2 norm error of $\bar{S}^2$ with changing grid points in all directions: $N_r(i) = 3^i = 3, 9, 27$, $N_t(i) = 4^i = 4, 12, 36$ and $N_z(i) = 4^i = 4, 12, 36$ for $i = 1, ..., 3$

as:

$$\frac{\partial \overline{u}_j}{\partial t} + \overline{u}_i \frac{\partial \overline{u}_j}{\partial x_i} = -\frac{1}{\rho}\frac{\partial \overline{p}}{\partial x_j} + \frac{\partial}{\partial x_i}\left([\nu + \nu_t]\frac{\partial \overline{u}_j}{\partial x_i}\right). \tag{7.20}$$

The Smagorinsky model has few drawbacks. The most important one is the behaviour of the SGS stress terms near the wall. The SGS stresses should be zero at the wall, and yet this is not guaranteed with this model. This shortcoming of the model can be eliminated by reducing the length scale l_S towards the wall with the usage of van Driest's damping function:

$$l_S = C_S\Delta\left[1 - exp\left(-\frac{y^+}{A^+}\right)^{n_1}\right]^{n_2}. \tag{7.21}$$

Here y^+ denotes the wall distance in viscous wall units ($y^+ = yu_*/\nu$), A^+ is a constant value of 26 and the powers n_1 and n_2 are both equal to unity (i.e., $n_1 = n_2 = 1$).

[5] Slope of 2 is expected for the given discretization scheme.

The computational domain used for LES is not the same one used for DNS. The only difference is the pipe length ($L = 2.5D$) which is relatively longer than the one considered in DNS ($L = D$). The reason is that the computational complexity of DNS is very high compared to LES (approximately a factor of 5). Consequently, in order to obtain more accurate results the pipe length considered for LES is chosen to be longer.

7.2.2 Results of LES computations

case	N_r	N_θ	N_z	y^+_{min}	C_S	$\langle u_z\rangle_c/u_*$	$\langle u_z\rangle_b/u_*$	$\langle u_z\rangle_c/\langle u_z\rangle_b$
1	16	64	64	32.81	0.12	26.73	20.23	1.32
2	16	64	64	32.81	0.10	25.27	20.23	1.24
3	16	64	64	9.42	0.10	25.52	20.23	1.26

Table 7.2: Mean flow quantities obtained by LES for the given Reynolds numbers $Re_c = \langle u_z\rangle_c D/\nu = 50500$, $Re_b = \langle u_z\rangle_b D/\nu = 42500$, $Re_* = u_* D/\nu = 2100$

The Smagorinsky model is used in all three cases and the Smagorinsky coefficient C_S is assigned a larger value for case 1 than the other cases as shown in Table 7.2. The results of Eggels [1] for $C_S = 0.10$ and for $C_S = 0.12$ are $\langle u_z\rangle_c/u_* = 24.08$, $\langle u_z\rangle_b/u_* = 19.64$, $\langle u_z\rangle_c/\langle u_z\rangle_b = 1.23$ and $\langle u_z\rangle_c/u_* = 24.80$, $\langle u_z\rangle_b/u_* = 20.28$, $\langle u_z\rangle_c/\langle u_z\rangle_b = 1.22$ respectively. The only difference between case 1 and 2 is the C_S value, which decreases to 0.10 in case 2. The wall damping function (Eq. (7.21)) is utilized in all the simulations, although the first grid point is not located close to the wall in all cases except case 3. The grid spacing is non-uniform (with a grid refinement factor of 1.15) in the radial direction for case 3, whereas the grid spacing is still uniform in circumferential and streamwise directions. The numerical resolution remains the same in case 3 compared to case 1 with C_S [6] unchanged, but the value of the first grid point decreases and it is now located at $y^+ = 9.42$.

Figures 7.6 and 7.7 illustrate the initial and final mean velocity profiles normalized with respect to the center line velocity for case 1 and 3 respectively. Fig. 7.8 compares the mean velocity profiles at different times. The turbulent fluctuations almost diminish after $t/t^* = 9$. In Fig. 7.9, the normalized mean velocity profiles are given as a function of the distance from the center line. The resulting mean velocities for the case 3 is almost identical to case 1 for all r/R except the near-wall region where smaller eddies are captured as the near-wall region resolved with a better numerical resolution with the use of non-uniform grid spacing. The increase in the Smagorinsky coefficient in case 2 does not have a significant influence on the profile in the inner region, however the resulting profile differs from $r/R = 0.35$ to the pipe wall (other than the near-wall region, both case 1 and 2 have similar profiles as expected). In contrast to DNS, the logarithmic region is well resolved in the numerical simulations of LES since the Reynolds number is much larger. The mean axial velocity profiles of all cases are shown in Fig. 7.10. The profiles show a logarithmic velocity distribution between $30 < y^+ < 300$. The law of the wall is only used in LES computations for validation purposes. The application of the wall function

[6] Please refer to Eq. (B.2) in Appendix B for a detailed explanation of the Smagorinsky constant C_S and l/l_f.

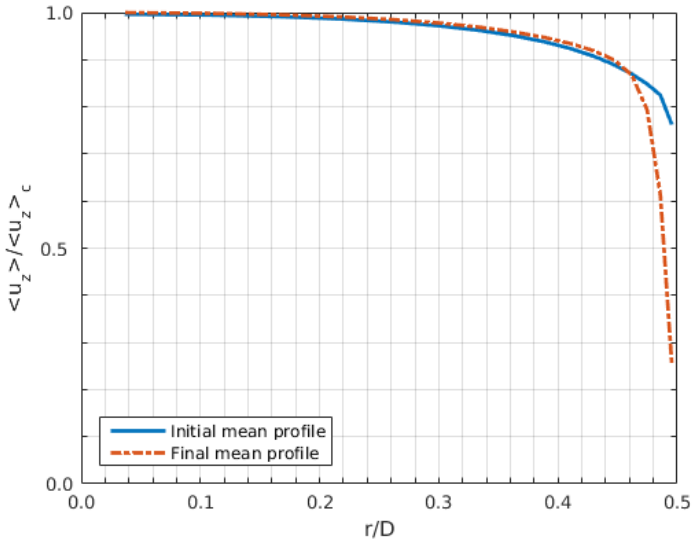


Figure 7.6: Initial normalized mean velocity and final normalized mean velocity profiles with respect to the center line velocity $\langle u_z \rangle_c$ as a function of the distance from the center line for LES-case 1

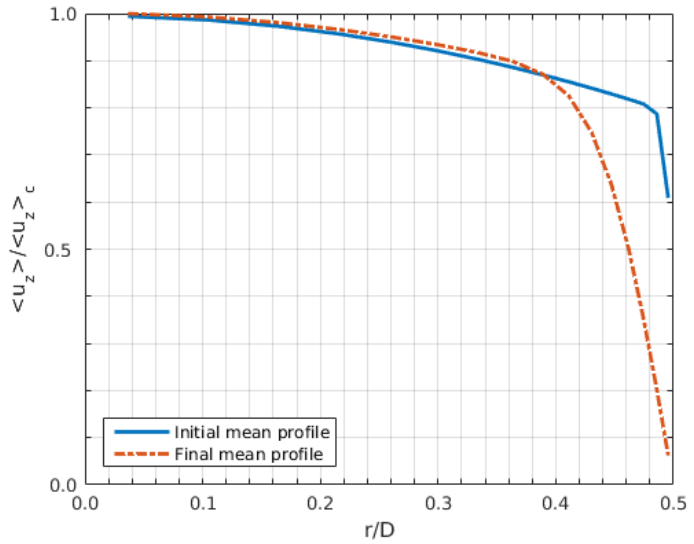


Figure 7.7: Initial normalized mean velocity and final normalized mean velocity profiles with respect to the center line velocity $\langle u_z \rangle_c$ as a function of the distance from the center line LES-case 3

only defines the first grid point from the wall, i.e. at $y^+ = y^+_{min}$. The development of the rest of the profile is achieved only by the governing equations. The results presented (the results of case 1 and 2) indicate that the LES computations are not very sensitive to small changes ($\pm 20\%$) of the Smagorisnky constant C_S. The optimum value of the Smagorinsky constant for the turbulent flow considered in this thesis is approximately

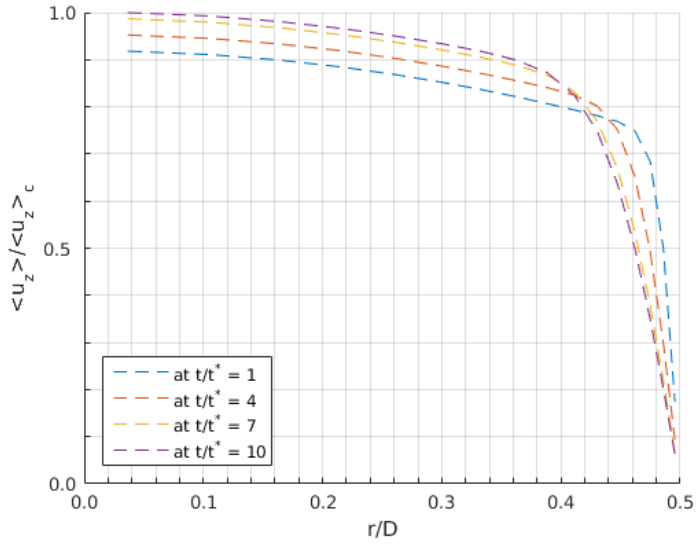


Figure 7.8: Development of mean velocity profiles nondimensionalized by the center line velocity $\langle u_z \rangle_c$ as a function of the distance from the center line through time for LES-case 2

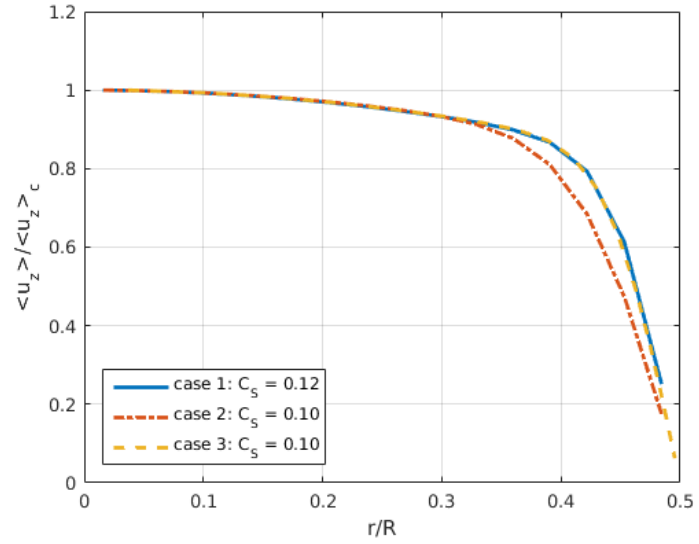


Figure 7.9: Mean axial velocity normalized on the center line velocity $\langle u_z \rangle_c$ as a function of the distance from the center line for $Re_c = 50500$

equal to 0.10. The results become unrealistic for larger or smaller values of C_S. Fig. 7.11 clearly demonstrates the aberration of case 2 from case 1. It also shows that case 2 is able to represent the near-wall region and the whole flow accurately with only 16 number of grid points in the radial direction.

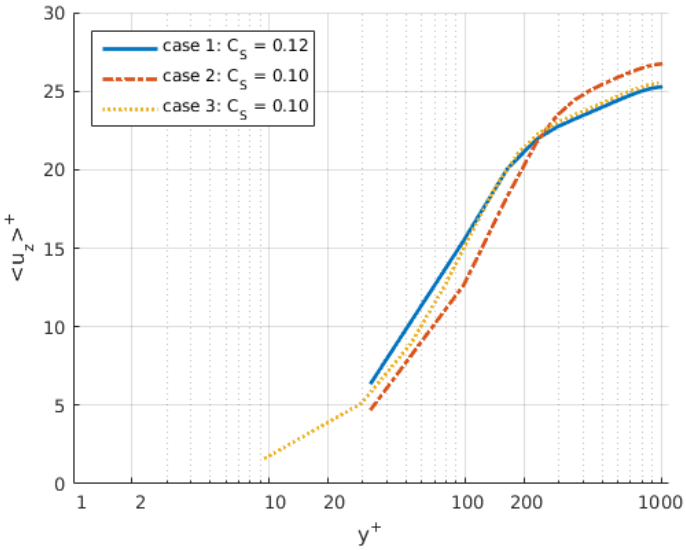


Figure 7.10: Mean axial velocity normalized by the friction velocity u_τ with respect to the distance from the wall (in wall units)

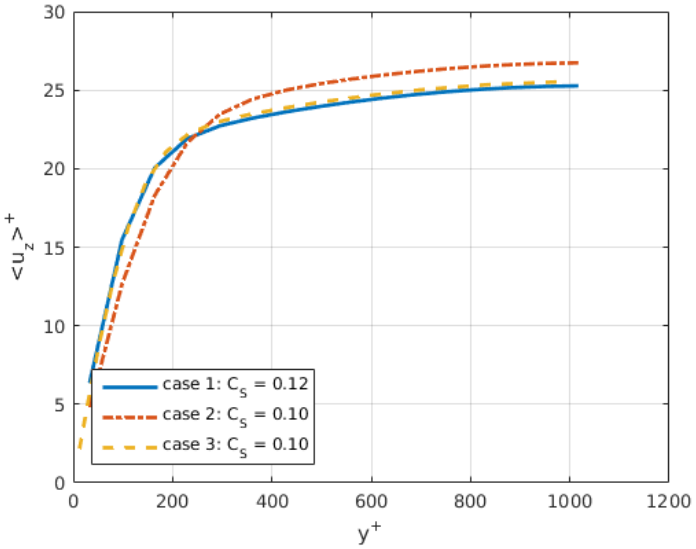


Figure 7.11: Mean axial velocity normalized by the friction velocity u_τ with respect to the distance from the wall (in wall units) for $Re_* = 2100$

In general, the results show that the Smagorinsky model is able to solve for the fully developed turbulent pipe flow problem. However, it is also stated in the study of Eggels [1] that the SGS parametrization does not yield results in good agreement with the experimental data in the near-wall region. The reason for this is the imperfect Smagorinsky model which does not allow for backscatter, requires a damping function in the near-

wall region and *a priori* input of the Smagorinsky constant. Although, the uniform grid spacing in the radial direction does not guarantee the capturing of sharp gradients in mean quantities near the pipe wall, the grid spacing in radial direction is kept uniform in cases 1 and 2 due to the fact that the wall damping function is utilized together with the conventional Smagorinsky model. The results of the simulations with the non-uniform grid spacing in radial direction are also in agreement with experimental results in the near-wall region. On the other hand, the grid spacing in streamwise and circumferential directions are relatively coarse and this may also lead to insufficient approximations. Furthermore, the dynamic SGS model (the dynamic Smagorinsky model) does not possess the shortcomings of conventional SGS turbulence model. Hence, it is a promising model for implementations in future.

Chapter 8

Conclusions and recommendations

In this thesis, modeling of immiscible incompressible turbulent two-phase stratified flow is planned to be investigated. For that reason, the most appropriate turbulent model for this type of pipe flow is elucidated by getting more insight into the current state-of-the-art of modeling of turbulent two-phase stratified flows. However, results for turbulent two-phase flows are not realized due to time limitations. Thus, the computations are only carried out for turbulent single-phase pipe flows using DNS and LES techniques.

First, estimates for the computational complexity in terms of degrees of freedom for DNS and LES are calculated and compared to differentiate the most appropriate model. By estimating the number of total grid points required for turbulent single-phase flows, which gives a rough estimate about the number of unknowns, a minimum requirement and an inference about the complexity of the turbulent two-phase flow are acquired.

The estimations of the required number of grid points for single-phase flow yield that both DNS and LES are possible. However, DNS is just within the limits of available computational power, whereas LES is on the safe side. In addition to this, the comparison of computational costs also revealed that DNS is only feasible for low Reynolds number turbulent single-phase pipe flows (considering the available computational resources and the lacking of parallel code in the Scientific Computing group of the Delft Institute for Applied Mathematics). Consequently, it can be stated that LES is more applicable than DNS in terms of computational complexity because of the coarser numerical resolution of LES.

The only differences between two-phase and single phase flow equations are the variable density and viscosity formulations. Other than that the equations for both cases are identical to each other provided the effects of surface tension are negligible and the influence of the interface and the momentum transfer between both phases are ignored. However, it will take a significant amount of time and effort to simulate the whole computational domain. For that reason, the domain length is reduced to decrease the computational complexity of the problem for DNS. On the other hand, LES is practical for both single

phase and turbulent two-phase flows without any restrictions. Besides, LES is also the superior approach for getting accurate results with less efforts and limited computational resources for high Reynolds number turbulent two-phase flows.

A variable viscosity formulation (the Smagorinsky model) is implemented to the algorithm together with the periodic boundary conditions both in circumferential and axial directions. With the help of the boundary condition the pressure difference over the pipe length is solved for simultaneously with the velocity profile as part of the problem in all numerical simulations. Furthermore, in order to realize the effect of the non-uniform grid spacing in the near-wall region, an additional case is performed and it is shown that the results of this case are in good agreement both in the near-wall and inner regions. As a minor drawback, it is worthwhile to mention that the required number of iterations for `GMRES` increases in the non-uniform case. The grid spacing in all other numerical simulations are uniformly spaced except that additional case in LES due to the wanting in *universality* of SGS parametrization. The results of the numerical simulations are validated with the study of Eggels [1]. In general, the statistical results are in good agreement with the study. Moreover, both DNS and LES are capable of simulating the problem within a reasonable amount of time and accuracy. It is also shown that choosing a relatively smaller pipe length (because of the restrictions imposed by the serial code) than the one chosen by Eggels [1] has no significant effect on the resulting velocity profile.

Further research can be carried out within the scope of this thesis as a future work. In order to obtain a rapid convergence rate, a faster and more robust preconditioner needs to be developed for the pressure solver. This will also allow the possibility of running DNS computations with much finer numerical resolutions. Also, the computational domain in axial direction (the pipe length L) needs to be much longer than the integral length scale of turbulent motions (i.e. L needs to be at least two and a half times longer than D). This approach is precluded in this thesis due to restrictions in the available computational power and lacking of a parallel code. Moreover, for LES computations non-uniform grid spacing approach can be realized with a larger number of grid points in the radial direction to increase the order of accuracy (especially in the near wall region) of the numerical results. In addition to this, rather than using the most simple model for the LES computations, i.e. the conventional Smagorinsky model, more advanced models can be utilized to perform better in the near-wall region.

After checking the feasibility of LES for turbulent two-phase flows, it is recommended to simulate immiscible incompressible turbulent two-phase stratified pipe flow problems using both DNS and LES methods. Additionally, the effect of the momentum transfer between two phases can be neglected in order to make the problem slightly easier since there is not any robust method to simulate turbulent two-phase stratified flow. The last but not least recommendation would be predicting the formation of slugs using VLES as suggested in [22], which predicts accurately the onset of the transition from wavy to slug flow. VLES can be used to decrease the required computational resources for the simulations (it minimizes the modelling efforts of interface dynamics and turbulence). It is also able to predict the turbulence in large-scale and capture transient motions of interfaces.

Appendix A

Estimating the computational cost of DNS

First, the estimation procedure, which has been carried out in the study of [1] for the total number of grid points is validated in this thesis for $Re_\tau = 395$. The following relations have been defined in the work of [1]:

$$L = 5D, \quad \tilde{L} = 5, \tag{A.1}$$

$$\tilde{l} = \frac{1}{10}\tilde{L} = 0.5, \tag{A.2}$$

where L is the pipe length, D is the diameter of the pipe, l is the largest length scale and the tilde symbol ˜ indicates a normalization in terms of the characteristic length scale (e.g., pipe diameter).

In the study, DNS has been carried out using $96 \times 128 \times 256$ grid points *equally* spaced in r-, θ-, and z-direction respectively. The grid spacing has been computed as $\Delta r^+ \approx 1.88$, $\Delta z^+ \approx 7.03$ in terms of viscous wall units [1]. The azimuthal grid spacing varies linearly with r and it has its minimum value $(\Delta r \Delta\theta/2)^+ \approx 0.05$ near the centerline of the pipe and its maximum value $(D\Delta\theta/2)^+ \approx 8.84$ at the wall [2].

The dimensionless grid spacing in r-direction is equal to

$$\tilde{h_r} = \frac{\Delta r}{D} = \frac{1}{2N_r}. \tag{A.3}$$

The number of grid points in r-direction is $N_r = 96$. Therefore,

$$\tilde{h_r} = \frac{1}{2(96)} \approx 0.0052. \tag{A.4}$$

[1] A superscript $^+$ corresponds to a normalization in terms of viscous wall units

[2] The maximum value of the grid spacing in θ-direction is calculated by assuming $D - \Delta r \approx D$ (subtracting Δr from the pipe diameter represents distance of the center point of the last grid cell to the center of the pipe), and for the minimum value of the grid spacing, the distance of the first grid cell to the center of the pipe is $\Delta r/2$.

Furthermore, the dimensionless grid spacing h_r (with respect to the pipe diameter) can be also calculated with the following approach (where $y^+ = 1$), which is in agreement with the study:

$$\tilde{h_r} = \frac{\Delta r}{D} = 2(y^+)\sqrt{74}(Re_b^{-13/14}) = 2\sqrt{74}(5300^{-13/14}) \approx 0.0059, \tag{A.5}$$

where the Reynolds number with the bulk velocity is $Re_b = 5300$.

The dimensionless grid spacing in θ-direction can be calculated as follows:

$$\tilde{h_\theta} = \frac{\Delta\theta}{2} = \frac{2\pi}{2N_\theta} = \frac{\pi}{128} \approx 0.0245, \tag{A.6}$$

where the number of grid points in θ-direction is $N_\theta = 128$. It can be seen that the ratio of the dimensionless grid spacing in θ-direction and dimensionless grid spacing in r-direction is $\tilde{h_\theta}/\tilde{h_r} \approx 4.7$. Moreover, the ratio of grid spacings computed in the study of [1] also has the same ratio $(D\Delta\theta/2)^+/\Delta r^+ \approx 4.7$

The total number of grid points in the z-direction is $N_z = 256$:

$$N_z = 256 = \frac{\tilde{L}}{\tilde{h_z}}, \tag{A.7}$$

where $\tilde{h_z}$ is the dimensionless grid spacing in z-direction, and is equal to

$$\tilde{h_z} = \frac{\tilde{L}}{N_z} = \frac{5}{256} \approx 0.0195. \tag{A.8}$$

The ratio of the dimensionless largest length scale to the dimensionless grid spacing in z-direction is $\tilde{l}/\tilde{h_z} = 0.5/0.0195 \approx 25.6$. The ratio of dimensionless grid spacing in z-direction and dimensionless grid spacing in r-direction is $\tilde{h_z}/\tilde{h_r} = 0.0195/0.0052 \approx 3.75$, which is almost same as the ratio given by the study $\Delta z^+/\Delta r^+ = 7.03/1.88 \approx 3.74$.

The same approach is implemented in this thesis. However, the Reynolds number Re_D computed with bulk velocity and characteristic length (in this case it is the pipe diameter) has a different value, $Re_D = 3421$. Therefore, the dimensionless grid spacing in r-direction $\tilde{h_r}$ (with respect to the pipe diameter, and $y^+ = 1$) is

$$\tilde{h_r} = \frac{\Delta r}{D} = 2(y^+)\sqrt{74}(Re_D^{-13/14}) = 2\sqrt{74}(3421^{-13/14}) \approx 0.0089. \tag{A.9}$$

Hence, the number of grid points in r-direction is

$$N_r = \frac{1}{2\tilde{h_r}} = \frac{1}{2(0.0089)} \approx 56. \tag{A.10}$$

If the same relations are used, i.e., $\tilde{h_z}/\tilde{h_r} \approx 3.75$, and $\tilde{h_\theta}/\tilde{h_r} \approx 4.7$, then $\tilde{h_z} \approx 0.0333$, and $\tilde{h_\theta} \approx 0.0418$. In this way, the number of grid points required in θ- and z-direction can be calculated respectively:

$$N_\theta = \frac{\pi}{\tilde{h_\theta}} = \frac{\pi}{0.0418} \approx 75, \tag{A.11}$$

$$N_z = \frac{\tilde{L}}{\tilde{h_z}} = \frac{50}{0.0333} \approx 1501, \tag{A.12}$$

where the length of the pipe is taken as one-fourth of the whole length of the pipe, i.e., $L = 50D$, thereby, the dimensionless pipe length with respect to the pipe diameter is $\tilde{L} = L/D = 50$. According to the study of [1], the two-point correlation coefficients get small enough around $L = 2.5D$. If we choose our computational domain length L equal to $2.5D$, then the dimensionless length scale with respect to the diameter is $\tilde{L} = 2.5$, which yields

$$N_z = \frac{\tilde{L}}{\tilde{h_z}} = \frac{2.5}{0.0333} \approx 75. \tag{A.13}$$

The uniform grid spacing approach above mimics the estimation procedure that has used in the work of [1]. The DNS approach below is carried out (without reverse engineering the study of [1]) with non-uniform grid spacing in r-direction, and uniform grid spacing in θ- and z-direction. The radial grid spacing has a minimum value $\Delta r^+ \approx 1$ near the pipe wall and reaches a maximum value $\Delta r^+ \approx 5$ near the centerline of the pipe. The azimuthal grid spacing varies linearly with r and reaches a minimum value $(\Delta r \Delta\theta/2)^+ \approx 0.0258$ near the centerline of the pipe and its maximum value $(D\Delta\theta/2)^+ \approx 5$ at the pipe wall.The grid spacing in axial direction $\Delta z^+ \approx 10$.

In order not to exceed the available computational power (30×150×90), the number of grid points and the aspect ratio for the grid spacing in r-direction is selected accordingly. The number of grids in θ-direction is calculated as follows:

$$(D\Delta\theta/2)^+ = (r\Delta\theta)^+ = \left(r\frac{2\pi}{N_\theta}\right)^+ \approx 5 \quad \rightarrow \quad N_\theta \approx 248. \tag{A.14}$$

The number of grids in z-direction is calculated as follows:

$$(\Delta z)^+ = \frac{L}{N_z} \approx 10 \quad \rightarrow \quad N_z \approx 198, \tag{A.15}$$

where $L = 5D$ and $D = 395$ in dimensionless wall units. Thus, the number of grid points required for DNS is 11×248×198, and the aspect ratio is $n = 1.16613$ for the non-uniform grid spacing in r-direction. The maximum available computational power is actually exceeded (approximately 20%) with these number of grid points.

Appendix B

Estimating the computational cost of LES

The characteristic grid spacing Δ_{ch} is calculated from the grid spacing in all three directions:

$$\Delta = \sqrt{\frac{\Delta r^2 + r^2\Delta\theta^2 + \Delta z^2}{3}}, \tag{B.1}$$

where Δr, $r\Delta\theta$, and Δz are the grid spacings in radial, tangential, and axial direction respectively. The ratio of l_{mix} [1] and Δ is denoted by the Smagorinsky coefficient c_s, which is a measure for the numerical resolution, as:

$$c_s = \frac{l_{mix}}{\Delta}. \tag{B.2}$$

Values of c_s less than 0.165 give insufficient spatial resolution [1]. Moreover, the ratio of l_{mix} and the filter length l_f:

$$\frac{l_{mix}}{l_f} = \frac{(\frac{3}{2}\alpha_K)^{-3/4}}{2\pi} = 0.0825, \tag{B.3}$$

where α_K is the Kolmogorov constant and equals approximately 1.6. Hence, if the same relation holds for the filtering length and characteristic grid spacing $l_f \geq 2\Delta$ (see Eq. (2.52)), then the following relation is obtained for the mixing length of the SGS motions:

$$l_{mix} = 0.0825\Delta \quad \rightarrow \quad \frac{l_{mix}}{\Delta} = c_s \geq 0.165. \tag{B.4}$$

In the work of [1], the value chosen for c_s is approximately 0.1, which is less than 0.165. The reason for using smaller value is that larger values of c_s does not maintain resolved scale motions and it gives unrealistic results in the end. The value of c_s is dependent on the type of flow in terms of the way how turbulence is produced. In order to obtain the

[1] The length scale l_{mix} represents the mixing length of the SGS motions.

number of grids required in LES, the Eq. (B.4) substituted into the Eq. (2.52) (Δ_x has been replaced by Δ):

$$N_L = \frac{L}{l}\frac{l}{l_f}\frac{l_f}{\Delta} = \frac{L}{l}\frac{l}{l_f}\frac{c_s}{0.0825}. \tag{B.5}$$

The following relation is obtained by approximating $L/l \approx 10$ and $c_s \approx 0.1$ for shear-driven turbulent flow:

$$N_L \approx 12\frac{l}{\Delta}. \tag{B.6}$$

In the study, equally spaced $16 \times 64 \times 128$ grid points (N_r, N_θ, and N_z) have been used for three cases, and the viscous wall layer has not been resolved in these cases. Therefore, the first grid-point is located within the inertial sublayer (i.e., $y^+_{min} = 32.8 > 30$). The grid spacing in r, θ, and z-direction are $\Delta r^+ = 65.6$, $(D\Delta\theta/2)^+ = 103.1$, and $\Delta z^+ = 82.0$ respectively.

The dimensionless mesh width (with respect to the pipe diameter) in r-direction can be calculated as follows:

$$\tilde{h_r} = \frac{1}{2N_r} = \frac{1}{32} \approx 0.0312. \tag{B.7}$$

The value above can be validated by using Eq. (A.9) with $y^+ = 32.8$:

$$\tilde{h_r} = \frac{\Delta r}{D} = 2(y^+)\sqrt{74}(Re_D^{-13/14}) = 2(32.8)\sqrt{74}(40000^{-13/14}) \approx 0.0301. \tag{B.8}$$

With the same approach the dimensionless mesh width in θ- and z-direction is calculated:

$$\tilde{h_\theta} = \frac{\Delta\theta}{2} = \frac{\pi}{N_\theta}\frac{\pi}{64} \approx 0.0490, \tag{B.9}$$

$$\tilde{h_z} = \frac{\tilde{L}}{N_z} = \frac{5}{128} \approx 0.0390. \tag{B.10}$$

The ratio of $\tilde{h_z}$ and $\tilde{h_r}$ equals to 1.25, which can be validated by taking the ratio of $\Delta z^+/\Delta r^+ \approx 1.25$. Moreover, the ratio of $\tilde{h_\theta}$ and $\tilde{h_r}$ approximately equals to 1.57. This ratio can also be validated by taking the ratio of $(D\Delta\theta/2)^+$ and Δr^+, which is also around 1.57.

In this thesis, the dimensionless mesh width in r-direction is calculated in Eq. (A.9) with $y^+ = 1$. For $y^+ = 32.8$, the dimensionless mesh width in r-direction is

$$\tilde{h_r} = \frac{\Delta r}{D} = 2(y^+)\sqrt{74}(Re_D^{-13/14}) = 2(32.8)\sqrt{74}(3421^{-13/14}) \approx 0.2919. \tag{B.11}$$

Thus, the number of grid points in r-direction is

$$N_r = \frac{1}{2\tilde{h_r}} = \frac{1}{0.5838} \approx 2 \tag{B.12}$$

It can be concluded that the boundary layer modeling is not feasible for this case since the Reynolds number is not large enough and the modeled boundary layer (viscous wall region) is almost equal to the radius of the pipe diameter. For other directions, the same approach can be implemented by using the ratios:

$$\tilde{h_\theta} = 1.57\tilde{h_r} \approx 0.4582, \tag{B.13}$$

$$\tilde{h_z} = 1.25\tilde{h_r} \approx 0.3648. \tag{B.14}$$

The number of grid points in θ- and z-direction can be calculated as follows:

$$N_\theta = \frac{\pi}{\tilde{h_\theta}} = \frac{\pi}{0.4582} \approx 7, \tag{B.15}$$

$$N_z = \frac{\tilde{L}}{\tilde{h_z}} = \frac{50}{0.3648} \approx 137, \tag{B.16}$$

where $\tilde{L} = 50$. For $\tilde{L} = 2.5$, the number of grid points in z-direction is

$$N_z = \frac{\tilde{L}}{\tilde{h_z}} = \frac{2.5}{0.3648} \approx 7. \tag{B.17}$$

The other computation case in the work of [1] has been carried out using $32\times128\times192$ grid points with *non-uniform* grid spacing only in r-direction (*uniform* grid spacing in other directions). The value of the dimensionless wall unit for these computations is $y^+ = 1.5$. The averaged value for the grid spacing in r-direction is $(\Delta r^+)^* = 32.8$, and for other directions the values are $(D\Delta\theta/2)^+ = 51.5$, $\Delta_z^+ = 54.7$.

The aspect ratio used in the non-uniform cases can be evaluated. In order to do that, first, the total sum of each grid spacings in r-direction should be computed as follows:

$$(\Delta r^+)^* = \frac{\sum_i \Delta r_i^+}{N_r}, \quad \rightarrow \quad \sum_i \Delta r_i^+ = (\Delta r^+)^*(N_r) = (32.8)(32) = 1049.6. \tag{B.18}$$

The value for the aspect ratio with the calculated total sum of grid spacings is approximately equals to 1.16613. The same aspect ratio is used in this thesis as well, and the number of grid points needed in r-direction can be calculated by using the same value for the aspect ratio. The first grid-point is at $\tilde{h_{r1}} = 0.0135$ for $y^+ = 1.5$. The number of grid points required in r-direction is $N_r = 16$, which is calculated by using the aspect ratio and the first grid spacing $\tilde{h_{r1}}$.

The ratio of the grid spacing in θ to r-direction is $(D\Delta\theta/2)^+/(\Delta r^+)^* \approx 1.5701$. Therefore, the grid spacing in θ-direction is

$$\tilde{h_\theta} = (1.5701)(\tilde{h_r})^* = (1.5701)(0.0625) \approx 0.0981, \tag{B.19}$$

where $(\tilde{h_r})^*$ is the averaged grid spacing in r-direction. Therefore, the number of grid points required in θ-direction is $N_\theta = 32$.

The ratio of the grid spacing in z to r-direction is $\Delta z^+/(\Delta r^+)^* \approx 1.6676$. Thus, the grid spacing in z-direction is

$$\tilde{h_z} = (1.6676)(\tilde{h_r})^* = (1.6676)(0.0625) \approx 0.1042. \tag{B.20}$$

Hence, the number of grid points required in z-direction is $N_z = 480$ for $\tilde{L} = 50$, and for $\tilde{L} = 2.5$, the number of grid points required in z-direction is $N_z = 24$.

Appendix C

MATLAB codes

C.1 Poiseuille channel flow

```
% Berkcan Kapusuzoglu <berkcan1992@gmail.com>
% MSc Thesis
% Created: June 2016
% Modified: August 2016

%% Problem Decsription
% The physical problem consists of a pipe connected to two reservoirs.
% The pressure at the inlet is higher than the pressure at the outlet
% reservoir, which  gives rise to a flow through the pipe from inlet
% to outlet. The velocity profile is assumed to be fully developed at
% all locations along the pipe and the pressure difference between left
% and right boundaries is constant. First, the problem is considered with
% an assumption that the computational domain is two-dimensional, in which
% the horizontal (normal) velocity only varies in vertical direction and
% vertical (tangential) velocity does not change. In order to implement
% the boundary conditions (i.e., periodic and no-slip boundary conditions),
% necessary modifications are made within the code.

%% Initialization of the variables
clc;clear;
rho = 997;                          % density of the fluud
mu= 1e-3;                           % dynamic viscosity of the fluid on bottom
mua= 1.85e-5;                       % dynamic viscosity of the gas on top
r=25e-3; l=10*r;                    % Radius and Length of the domain
lx = l; ly = 2*r;                   %length of the domain x- and y-directions
nh=3;
nxx=3;
dt = 1e-3;                         % time step in [s]
ny(1:nh)=0;
slop=1;iss(2)=0;ero(2)=0;cc(2)=0;m(1,nxx)=0;Zp(3,3)=0;
Zu(nxx,nxx)=0;Zuex(1,1)=0;rero(nh)=0;mmu(1,1)=0;mmuex(1,1)=0;
mm(3,3)=0;error(nh,:)=0;errorp(nh,:)=0;hx(nh)=0;hy(nh)=0;
is = 0;vtol = 1;stol = 1;
ind = floor(nxx^(nh-1)/2)+1;

```

```
% # grid points in x- and y-directions
ny = 196;                          % 3^ih = 3,9,27
nx = 16;                           % 3^ih = 3,9,27
hx = lx/(nx);                      % mesh width in x-dir
hy = ly/(ny);                      % mesh width in y-dir
x = linspace(0,l-hx,nx);           % edge nodes locations in x-dir
y = 0+(hy/2):hy:(ly)-(hy/2);       % edge nodes locations in y-dir
xp = linspace(0+hx/2,l-hx/2,nx);% center nodes locations in x-dir
yp = 0+(hy/2):hy:(ly)-(hy/2);      % center nodes locations in y-dir
lxx=xp(1,nx)-xp(1,1);              % distance between first
                                   % and last pressure nodes

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%% Calculate the Coefficient Matrices %%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% R, M, D, S, G, gcc representing the discretization
%%% of time derivative, the continuity equation. the
%%% viscous terms, the pressure term, and the unknown
%%% constant pressure jump c.


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%% VECTOR R %%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
R = sparse(1,1:nx:nx*ny,1,1,nx*ny);
R = hy*R;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%% MATRIX M %%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
M = sparse(1:nx*ny,1:nx*ny,1);
Mh = hx*hy*rho*M;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%% MATRIX D %%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

D = gallery('tridiag',nx*ny,0,-1,1);
    for i=1:ny;
        D(nx*i,nx*(i-1)+1) = 1;
        D(nx*i,nx*(i)+1) = 0;
    end
D(:,nx*ny+1) = [];
D = (1/hx)*D;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%% MATRIX S %%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
s1 = sparse(1,1:nx*ny,-3);
s2 = sparse(1,1:nx*(ny-1),1);
s3 = sparse(1,1:nx*(ny-1),1);
sd1 = diag(s1, 0) ;
sd2 = diag(s2, -nx) ;
sd3 = diag(s3, nx);

% For the inner cells of -2 entry rather than -3
for i=1:(ny-2)
    for j=1:nx
        sd1(nx*i+j,nx*i+j) = -2;
    end
end
Si = sd1+sd2+sd3;
```

```
98
99  S(1:nx*ny/2,:) = Si(1:nx*ny/2,:)*mu*(1/(hy^2));
100 S(nx*ny/2+1:nx*ny,:) = Si(nx*ny/2+1:nx*ny,:)*mua*(1/(hy^2));
101 Sf = full(S);
102
103 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
104 %%%%%%%%%%%%%%%%%%%%%% MATRIX G %%%%%%%%%%%%%%%%%%%%%%%%%%%%
105 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
106 G = gallery('tridiag',nx*ny,-1,1,0);
107     for i=1:ny-1;
108         G(1+nx*(i-1),nx*i) = -1;
109         G(nx*i+1,nx*i) = 0;
110     end
111 G(nx*(ny-1)+1,nx*ny) = -1;
112 Gh = (1/hx)*G;
113 % Gh = G;
114
115 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
116 %%%%%%%%%%%%%%%%%%%%%% MATRIX gcc %%%%%%%%%%%%%%%%%%%%%%%%%%
117 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
118 Gc = sparse([1:nx:nx*ny],1,-1,nx*ny,1);
119 gcc = (1/hx)*Gc;
120 % gcc = Gc;
121
122 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
123 %%%%%%%%%%%%%% CALCULATION OF THE NS EQ %%%%%%%%%%%%%%%%%%%%
124 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
125 % Initialize velocity and pressure vectors
126 u = zeros(nx*ny,1);
127 p = zeros(nx*ny,1);
128 c = 0;
129
130 qv1 = 1e-3;
131 dpdx1 = -12*mu*qv1/ly^3;
132 vmax = - (ly^2/(8*mu)) * dpdx1;
133 % vavg = -(ly^2)*dpdx1/(32*mu);
134 % delp = (vmax * 4 *mu/rho)/(r^2);
135 % qv = -(delp*pi*(r^4))/(8*mu);
136
137 for i=1:ny;
138     for j=1:nx
139         % BC for inlet velocity
140         u((i-1)*(nx)+j,1) = - (ly*((ly-hy/2)-hy*(i-1))/(2*mu))*dpdx1...
141         *(1-(((ly-hy/2)-hy*(i-1))/(ly)));
142     end
143 end
144 u(nx*ny/2+1:nx*ny)=u(nx*ny/2+1:nx*ny)*mu/mua;
145 % uex = u;
146 ut = u;
147 % qv = R*ut;      % qv =0.001
148
149 % PRESCRIBED FLOW RATE
150 qv = 5e-3;
151 dpdx2 = -12*mu*qv/ly^3;
152 vmax2 = - (ly^2/(8*mu)) * dpdx2;
153
154 % INITIALIZATION OF THE PROBLEM
155
156 %   Laplacian type Eq. 23b
157     lap1 = Mh\gcc;
158     lap2 = R*lap1;
159     lap3 = Mh\Gh;
```

```
160      lap4 = gcc*(R*lap3);
161      lap5 = lap4/lap2;
162      lap6 = Gh-lap5;
163      lap7 = Mh\lap6;
164      lap8 = D*lap7;
165
166      pf1 = (D*ut)/dt;
167      pf2 =(D*lap1)/(lap2);
168      pf3 = (qv-R*ut)/dt;
169      b = pf1 + (pf2*pf3);
170
171      [pf,flag,relres] = minres(lap8,b,1e-8,60);
172      p = p + pf;
173
174  %     Constant c, Eq. 23a
175      cf = -(1/(lap2))*(R*(Mh\(Gh*pf)) + pf3);
176      c = c + cf;
177
178      %     Eq. 20
179      alf = -Gh*pf - gcc*cf;
180      u = (Mh\(alf))*dt + ut;
181
182      % introduce the matrix for the implicit scheme
183      spy=speye(nx*ny,nx*ny);
184      uimp = spy - (Mh\S)*dt;
185
186  while abs(vtol)>1e-7 || abs(stol)>1e-7
187
188      is = is + 1;
189      utol = u;
190      ptol = p;
191      slop1 = slop;
192
193  % % Temporary Velocity Eq. 19  for the explicit method
194  %     alf = S*u - Gh*p - gcc*c;
195  %     ut = u + (Mh\alf)*dt;
196
197      % Temporary Velocity Eq. 19  implicit
198      alf = - Gh*p - gcc*c;
199      ut = uimp\(u + (Mh\alf)*dt);
200
201  %   Laplacian type Eq. 23b
202      pf1 = (D*ut)/dt;
203      pf3 = (qv-R*ut)/dt;
204      b = pf1 + (pf2*pf3);
205
206      [pf,flag,relres] = minres(lap8,b,1e-7,100);
207      p = p + pf;
208
209  %     Constant c, Eq. 23a
210      cf = (-1/(lap2))*(R*(Mh\(Gh*pf)) + pf3);
211      c = c + cf;
212
213  %     Eq. 20
214      alf = -Gh*pf - gcc*cf;
215      u = (Mh\(alf))*dt + ut;
216
217  for i=1:nx
218      m(1,i) = p(i,1);
219  end
220
221  iss(is)=is;
```

```
slop=-(m(1,1)-m(1,nx))/lxx;
stol = slop-slop1;
vtol = u((floor(ny/2))*(nx)+1,1)-utol((floor(ny/2))*(nx)+1,1);
cc(is)=c;
end
vmax1=u((floor(ny/2))*(nx)+1,1);

for j=1:ny
    for i=1:nx
        Zu(i,j) = u((j-1)*(nx)+i,1);
        Zp(i,j) = p((j-1)*(nx)+i,1);
     end
end

[X,Y] = meshgrid(x,y);
[Xp,Yp] = meshgrid(xp,yp);


 figure(1);
 Zr=rot90(Zu,1);
 contourf(X,Y,Zr);
 colormap jet
 h=colorbar('SouthOutside');
 % set(h, 'Position', [.125 .5 .7950 .03]);
 set( h, 'XDir', 'reverse' );
 xlabel('Length [m]','fontsize',12);
 ylabel('Height [m]','fontsize',12);
 title('Velocity Profile','fontweight','bold','fontsize',16);

 figure(2);
 plot(Zr, Y);
 % axis([0 vmax1 + 0.01 0 ly])
 ylabel('Height [m]','fontsize',12);
 xlabel('Velocity [m/s]','fontsize',12);

 Zpp=rot90(Zp,-3);
 figure(3);
 contourf(Xp,Yp,Zpp);
 colormap jet
 h=colorbar('SouthOutside');
 % set(h, 'Position', [.125 .5 .7950 .03]);
 set( h, 'XDir', 'reverse' );
 xlabel('Length [m]','fontsize',12);
 ylabel('Height [m]','fontsize',12);
 title('Pressure Profile','fontweight','bold','fontsize',14);

 figure(4);
 h(1)=plot(xp,m,'r');
 legend(sprintf('Slope = %f',slop(1)) , 'Location', 'NorthEast')
 hold on;
 plot(Xp, Zpp,'Marker','o');
 ylabel('Pressure [kg.m^{-1}.s^{-2}]','fontsize',12);
 xlabel('Length [m]','fontsize',12);
 title('Pressure Drop','fontweight','bold','fontsize',14);
 hold off;

 figure(5);
 plot(iss,ptol);
 set(gca, 'XScale', 'log')
 grid on
 xlabel('Time steps','fontsize',12);
 ylabel('Pressure drop ','fontsize',12);
```

```
284
285   figure(6);
286   plot(iss,cc);
287   set(gca, 'XScale', 'log')
288   grid on
289   xlabel('Time steps','fontsize',12);
290   ylabel('Pressure constant c','fontsize',12);
291
292  % Slope value
293   txt1 = ['       \leftarrow Slope = ' num2str(slop(nstep(ih)))];
294   text(xp(1,2),m(1,2),txt1)
295   legend(h(1), 'Slope' , 'Location', 'NorthEast')
296   plot(xp,m1,'b')
297   plot(xp,m2,'g')
298
299   figure(5);
300   scatter(h,rero);
301   slope7 = (log(error(3))-log(error(2)))/(log(hy(3))-log(hy(2)));
302   loglog(hx/ly,errorp,'r');
303   hold on;
304   refline(2);
305   legend(sprintf('Slope = %f',slope7), 'Slope = 2' , 'Location', ...
306  'NorthEast')
307   'Location', 'NorthEast' );
308   grid on;
309
310   figure(7);
311   scatter(hy/ly,error);
312   set(gca, 'YScale', 'log')
313   set(gca, 'XScale', 'log')
314   b = polyfit(log(hy/ly), log(error'), 1);
315   wetfit = exp(b(2)) .* (hy/ly).^b(1);
316   hold on
317   ph1 = plot(hy/ly, wetfit,'r');
318   hold off
319   hold on
320   ph3=refline(2);
321   legend( [ph1 ph3],{sprintf('Slope = %f',b(1)),'Slope = 2'},...
322   'Location', 'NorthEast' );
323   hold off;
324   grid on;
325   xlabel('hy/D','fontsize',12);
326   ylabel('Inlet Velocity L2-Norm','fontsize',12);
327
328   figure(8);
329   scatter(hy/ly,errorp);
330   set(gca, 'YScale', 'log')
331   set(gca, 'XScale', 'log')
332   b = polyfit(log(hy/ly), log(errorp'), 1);
333   wetfit = exp(b(2)) .* (hy/ly).^b(1);
334   hold on
335   ph1 = plot(hy/ly, wetfit,'r');
336   hold off
337   hold on
338   ph3=refline(2);
339   legend( [ph1 ph3],{sprintf('Slope = %f',b(1)),'Slope = 2'},...
340   'Location', 'NorthEast' );
341   hold off;
342   grid on;
343   xlabel('hy/D','fontsize',12);
344   ylabel('Pressure Gradient L2-Norm','fontsize',12);
345
```

```
%%%%%% PRINT OUT %%%%%
fprintf('The volumetric flow rate initially: Ru* = %d \n', qv1);
fprintf('The imposed volumetric flow rate: Q = %d [m^2/s] \n', qv);
fprintf('The volumetric flow rate finally: Ru* = %d [m^2/s] \n \n', R*ut);
fprintf(['Max. velocity calculated by Hagen-Poisouille Eq. for initial\n'...
'flow rate %d [m^3/s] is = %d [m/s] \n'],qv1,vmax);
fprintf(['Max. velocity calculated by Hagen-Poisouille Eq. for final\n'...
    'flow rate %d [m^3/s] is = %d [m/s] \n'],qv,vmax2);
fprintf(['Max. velocity after %d iterations and total time %d [s] =\n'...
    '%d [m/s] \n\n '], is,is*dt, vmax1);
fprintf(['Initial theoretical pressure gradient calculated after %d'...
    'iterations \n by Hagen-Poisouille Eq. = %d \n'], is, dpdx1);
fprintf(['Final theoretical pressure gradient calculated after %d'...
    'iterations \n by Hagen-Poisouille Eq. = %d \n'], is, dpdx2);
fprintf('Slope calculated numerically = %d \n\n', slop);
fprintf('||p^(n+1)-p^(n)|| = %d \n\n', ((sum((pf).^2))^0.5));
fprintf('The value for c = %d \n', c);
fprintf('|c^(n+1)-c^(n)| = %d \n\n', abs(cf));
re = (vmax2*2/3)*ly*rho/mu
```

C.2 2D Poiseuille axisymmetric pipe flow

```
% Berkcan Kapusuzoglu <berkcan1992@gmail.com>
% MSc Thesis
% Created: June 2016
% Modified: August 2016

%% Problem Decsription
% The phys ical problem consists of a pipe connected to two reservoirs.
% The pressure at the inlet is higher than the pressure at the outlet
% reservoir, which  gives rise to a flow through the pipe from inlet
% to outlet. The velocity profile is assumed to be fully developed at
% all locations along the pipe and the pressure difference between left
% and right boundaries is constant.

%% Initialization of the variables
clc;clear;
tic;
rho = 997;                      % density of the fluid [kg/m^3]
% rho1 = 1.1839;                  % density of the gas [kg/m^3]

r=25e-3; l=10*r;                % Radius and Length of the domain
lx = l; ly = 2*r;               % length of the domain x- and y-directions
slop=1;iss(2)=0;cc(2)=0;m(1,3)=0;Zp(3,3)=0;Zu(3,3)=0;rh(2)= 0;
rt(2)= 0;rhh(2)=0;R(2,2)=0;is = 0;vtol = 1;stol = 1;

fr_tol = 1e-6;
dt = 1e-2;                       % time step in [s]
% # grid points in x- and y-directions
nx = 196;
ny = 86;
nt = 10;
dtt = 2*pi/nt;
hx = lx/(nx);                   % mesh width in x-dir
hy = r/(ny);                    % mesh width in y-dir
x = linspace(0,l-hx,nx);        % edge nodes locations in x-dir
```

```
y = (hy/2):hy:r;     % edge nodes locations in y-dir
ya = -r+hy/2:hy:-(hy/2);
y = [ya y];
xp = linspace(0+hx/2,l-hx/2,nx);% center nodes locations in x-dir
yp = 0+(hy/2):hy:(r)-(hy/2);    % center nodes locations in y-dir
lxx=xp(1,nx)-xp(1,1);           % distance between first
                                % and last pressure nodes

b = 1e-3;                       % bottom viscosity
mu1 = 1.00e-3;               % top viscosity for constant mu
% mu1 = 1e-4;                     % top viscosity for variable mu
gam=log(mu1/b)/r;
a = (mu1 - b)/r;
mu = a*abs(y) + b;
% mu=b*exp(gam*y);

% Initialize velocity and pressure vectors
u = zeros(nx*ny,1);
p = zeros(nx*ny,1);
c = 0;

% PRESCRIBED FLOW RATE
% Initial flow rate
vmax = 0.13;
dpdx1 = -vmax/((r^2)/(4*b));
Q = -(pi*(r^4))*dpdx1/(8*b);

% Desired velocity for Re = 6950
vmax2 = 0.13941825;
dpdx2 = -vmax2/((r^2)/(4*b));
qv = -(pi*(r^4))*dpdx2/(8*b);

for ri=1:ny
    for iv=1:nx;
    % Full-developed Poiseuille flow velocity
    u((ri-1)*(nx)+iv,1)=vmax*(1-( (((hy/2)+hy*(ri-1))^2)/(r^2)) );
    uri(ri,1)=vmax*(1-( (((hy/2)+hy*(ri-1))^2)/(r^2)) );
    end
end
up(ny,1)=0;
for ri=1:ny;
    for iv=1:nx;
    % Full-developed Poiseuille flow velocity for ar+b
    us((ri-1)*(nx)+iv,1)= (dpdx2/(2*a^2)) * (a* (((hy/2)+hy*(ri-1))-r)...
        -b*log((a*((hy/2)+hy*(ri-1)) + b)/(a*r + b)));
    us1((ri-1)*(nx)+iv,1)= (dpdx1/(2*a^2)) * (a* (((hy/2)+hy*(ri-1))-r)...
        -b*log((a*((hy/2)+hy*(ri-1)) + b)/(a*r + b)));
    up(ri,1) = (dpdx1/(2*a^2)) * (a* (((hy/2)+hy*(ri-1))-r) ...
        -b*log((a*((hy/2)+hy*(ri-1)) + b)/(a*r + b)));
    up1(ri,1) = (dpdx2/(2*a^2)) * (a* (((hy/2)+hy*(ri-1))-r)...
    -b*log((a*((hy/2)+hy*(ri-1)) + b)/(a*r + b)));
    end
end
upf=fliplr(up');
up=[upf';up];
upf=fliplr(up1');
up1=[upf';up1];

urif=fliplr(uri');
uri=[urif';uri];

% figure (15)
```

```
% plot(up,y)
% hold on
% plot(uri,y)
% ylabel('Height [m]','fontsize',12);
% xlabel('Velocity [m/s]','fontsize',12);
% hold off
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%% Calculate the Radius Vector %%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% Distance of Nodes From the Center
% rh represents the distance from the centers of the CV (control volumes)
% from the center of the pipe.
% rt represents the distance from the edges of the CV from the
% center of the pipe.
for ri=1:ny;hold off

    % r value for u velocity
    % r1/2 values
    rh(ri) = abs(hy/2+hy*(ri-1));
    % r1 values
    rt(ri) = abs(hy*(ri));
    if ri==1
    % First part
    % u13 (velocity pont at 13th node, increase in axial then radial dir.)
    fr = -(a*rh(ri)+b)*rt(ri)/rh(ri);
    elseif ri==ny
    % Last part
    % u17 to u20
    l1 = (a*rh(ri-1)+b) * rt(ri-1)/rh(ri);
    % u21 to u25
    l2 = (-(a*rh(ri)+b)*2*rt(ri)-(a*rh(ri-1)+b)*rt(ri-1))/rh(ri);
    % for the top layer coming from ghost cell add additional rh
    else
    % Inner part
    % u13
    in1(ri) = (a*rh(ri-1)+b) * rt(ri-1)/rh(ri);
    % u17
    in2(ri) = (-(a*rh(ri)+b)*rt(ri) - (a*rh(ri-1)+b)*rt(ri-1))/rh(ri);
    % u21
    in3(ri) = (a*rh(ri)+b) * rt(ri)/rh(ri);
    end
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%% Calculate the Coefficient Matrices %%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% R, M, D, S, G, gcc representing the discretization
%%% of time derivative, the continuity equation. the
%%% viscous terms, the pressure term, and the unknown
%%% constant pressure jump c.


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%% VECTOR R %%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
R = sparse(1,1:nx:nx*ny,1,1,nx*ny);
Rr = R;
for ri=1:ny
  R(1,nx*(ny-1)+1-nx*(ri-1)) = nt*dtt*rh(ny-(ri-1))*hy...
      *Rr(1,nx*(ny-1)+1-nx*(ri-1));
end

```

```
159 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
160 %%%%%%%%%%%%%%%%%%%%%% MATRIX M %%%%%%%%%%%%%%%%%%%%%%
161 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
162 M = sparse(1:nx*ny,1:nx*ny,1);
163 Mh = rho*M;
164 % for ri=1:ny;
165 %     Mh = rho(ri)*M;
166 % end
167 % for ri=1:ny;
168 %     for i=1:nx;
169 %         M((ri-1)*nx+i,(ri-1)*nx+i) = 2*pi*rh(ri)*...
170 %             M((ri-1)*nx+i,(ri-1)*nx+i);
171 %     end
172 % end
173 % Mh = hx*hy*rho*M;
174
175 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
176 %%%%%%%%%%%%%%%%%%%%%% MATRIX D %%%%%%%%%%%%%%%%%%%%%%
177 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
178
179 D = gallery('tridiag',nx*ny,0,-1,1);
180     for ri=1:ny;
181         D(nx*ri,nx*(ri-1)+1) = 1;
182         D(nx*ri,nx*(ri)+1) = 0;
183     end
184 D(:,nx*ny+1) = [];
185 D = (1/hx)*D;
186
187 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
188 %%%%%%%%%%+%%%%%%%%%%%% MATRIX S %%%%%%%%%%%%%%%%%%%%%%
189 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
190 s1 = sparse(1,1:nx*ny,in2(2));
191 s2 = sparse(1,1:nx*(ny-1),in1(2));
192 s3 = sparse(1,1:nx*(ny-1),in3(2));
193
194 sd1 = diag(s1, 0) ;
195 sd2 = diag(s2, -nx) ;
196 sd3 = diag(s3, nx);
197
198 %for inner cells of -2 entry rather than -3
199 for iv=1:nx
200     sd1(nx*(ny-1)+iv,nx*(ny-1)+iv) = l2;
201     sd2(nx*(ny-1)+iv,nx*(ny-2)+iv) = l1;
202     sd1(iv,iv) = fr;
203     sd3(iv,nx+iv) = -fr;
204 end
205 for j=1:ny-3
206     for i=1:nx
207             sd1((j+1)*nx+i,(j)*nx+i) = in1(j+2)-in1(2);
208             sd1((j+1)*nx+i,(j+1)*nx+i) = in2(j+2);
209             sd3((j+1)*nx+i,(j+2)*nx+i) = in3(j+2);
210     end
211 end
212 sdf1=full(sd1);
213 S1 = sd1+sd2+sd3;
214 % Sf = full(S1);
215 S = S1*(1/(hy^2));
216
217 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
218 %%%%%%%%%%%%%%%%%%%% MATRIX G %%%%%%%%%%%%%%%%%%%%
219 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
220 G = gallery('tridiag',nx*ny,-1,1,0);
```

```
221 for ri=1:ny-1;
222     G(1+nx*(ri-1),nx*ri) = -1;
223     G(nx*ri+1,nx*ri) = 0;
224 end
225 G(nx*(ny-1)+1,nx*ny) = -1;
226 Gh = (1/hx)*G;
227 % Gf=full(G);
228 % Gh = G;
229
230 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
231 %%%%%%%%%%%%%%%%%%%% MATRIX gcc %%%%%%%%%%%%%%%%%%%%%%%%%%%%
232 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
233 Gc = sparse([1:nx:nx*ny],1,1,nx*ny,1);
234 % Gcr = sparse([nx:nx:nx*ny],1,1,nx*ny,1);
235 gcc = (1/hx)*Gc;
236 % gcct=(1/hx)*Gcr;
237 % Gch=full(Gc);
238 % gcc = Gc;
239
240 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
241 %%%%%%%%%%%%% CALCULATION OF THE NS EQ %%%%%%%%%%%%%%%%%%%%%
242 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
243 ut = u;
244 qvc = R*ut;
245
246 %% INITIALIZATION OF THE PROBLEM
247 % Laplacian type Eq. 23b
248     lap1 = Mh\gcc;
249     lap2 = R*lap1;
250     lap3 = Mh\Gh;
251     lap4 = gcc*(R*lap3);
252     lap5 = lap4/lap2;
253     lap6 = Gh-lap5;
254     lap7 = Mh\lap6;
255     lap8 = D*lap7;
256     pf1 = ((D*ut)/dt);
257     pf2 =(D*lap1)/(lap2);
258     pf3 = (qv-qvc)/dt;
259     b1 = pf1 + (pf2*pf3);
260     rhs1=-b1;
261
262 %% Preconditioner
263 lap9 = D*(Mh\Gh);
264 alpha = .1;
265 % L2 = ichol(-lap9, struct('type','ict','droptol',1e-7,'diagcomp',alpha));
266 L2 = ichol(-lap9);
267 [pf,fl1,rr1,it1,rv1]=gmres(lap8,b1,[],1e-10,100,L2,L2');
268 % figure(30)
269 % semilogy(0:it1(2),rv1/norm(b),'-o')
270 % xlabel('Iteration number');
271 % ylabel('Relative residual');
272 p = p + pf;
273 % p01=p(1);
274 % dpdz0=-(p(1)-p(nx))/lxx;
275 %     qvravg = -pf3/lap2;
276 %     pres = -(1/(lap2))*(R*(Mh\(Gh*pf)));
277
278     % Constant c, Eq. 23a
279     cf = -(1/(lap2))*(R*(Mh\(Gh*pf)) + pf3);
280     c = c + cf;
281
282     % Eq. 20
```

```
283      alf = -Gh*pf - gcc*cf;
284      u = (Mh\(alf))*dt + ut;
285
286      spy=speye(nx*ny,nx*ny);
287      uimp = (2*spy/dt - (Mh\S));
288  %      umf = full(uimp);
289
290  while(abs(pf3*dt/qv)>abs(fr_tol))
291      is = is + 1;
292
293  %      % Temporary Velocity Eq. 19  implicit
294      alf = - Gh*p - gcc*c;
295      rhsu=(2*u/dt + (Mh\alf));
296      [uf,¬,¬]=gmres(uimp,rhsu,[],1e-11,100);
297      ut=2*uf-u;
298
299      % Laplacian type Eq. 23b
300      pf1 = (D*ut)/dt;
301      pf3 = (qv-R*ut)/dt;
302      b1 = pf1 + (pf2*pf3);
303  %      figure(31)
304      [pf,fl1,rr1,it1,rv1]=gmres(lap8,b1,[],1e-10,100,L2,L2');
305      p = p + pf;
306      % Constant c, Eq. 23a
307      cf = (-1/(lap2))*(R*(Mh\(Gh*pf)) + pf3);
308      c = c + cf;
309
310      % Eq. 20
311      alf1 = -Gh*pf - gcc*cf;
312      u = (Mh\(alf1))*dt + ut;
313
314  for i=1:nx
315      m(1,i) = p(i,1);
316  end
317   iss(is)=is;
318  slop=-(m(1,1)-m(1,nx));
319  cc(is)=c;
320  end
321  vmax1=max(u);
322      for ri=1:ny
323          for iv=1:nx
324              Zu(iv,ri) = u((ri-1)*(nx)+iv,1);
325              Zp(iv,ri) = p((ri-1)*(nx)+iv,1);
326          end
327      end
328  [X,Y] = meshgrid(x,y);
329  [Xp,Yp] = meshgrid(xp,yp);
330  Zuu=rot90(Zu,2);
331  % Zr(ny+1,:) = 0;
332  Zuu=[Zuu Zu];
333  Zr=rot90(Zuu,3);
334
335   figure(1);
336   contourf(X,Y,Zr);
337   colormap jet
338   h=colorbar('SouthOutside');
339   set( h, 'XDir', 'reverse' );
340   xlabel('Length [m]','fontsize',12);
341   ylabel('Height [m]','fontsize',12);
342
343   figure(2);
344   plot(Zr, Y,'b');
```

```
345   hold on;
346   p1=plot(vmax2,x(1),'ro');
347   % axis([0 vmax1 + 0.01 0 ly])
348   legend(p1,sprintf('Maximum theoretical velocity = %f',vmax2), ...
349       'Location', 'NorthEast')
350   ylabel('Height [m]','fontsize',12);
351   xlabel('Velocity [m/s]','fontsize',12);
352   plot(up1,y,'r')
353   hold off;
354
355   Zpp=rot90(Zp,-3);
356   figure(3);
357   contourf(Xp,Yp,Zpp);
358   colormap jet
359   h=colorbar('SouthOutside');
360   % set(h, 'Position', [.125 .5 .7950 .03]);
361   set( h, 'XDir', 'reverse' );
362   xlabel('Length [m]','fontsize',12);
363   ylabel('Height [m]','fontsize',12);
364   % title('Pressure Profile','fontweight','bold','fontsize',14);
365
366   Zpf=rot90(Zpf,-3);
367   figure(10);
368   contourf(Xp,Yp,Zpf);
369   colormap jet
370   h=colorbar('SouthOutside');
371   % set(h, 'Position', [.125 .5 .7950 .03]);
372   set( h, 'XDir', 'reverse' );
373   xlabel('Length [m]','fontsize',12);
374   ylabel('Height [m]','fontsize',12);
375
376   figure(4);
377   h(1)=plot(xp,m,'r');
378   legend(sprintf('Slope = %f',slop(1)/lxx) , 'Location', 'NorthEast')
379   hold on;
380   plot(Xp, Zpp,'Marker','o');
381   ylabel('Pressure [kg.m/s^{2}]','fontsize',12);
382   xlabel('Length [m]','fontsize',12);
383   title('Pressure Drop','fontweight','bold','fontsize',14);
384   hold off;
385
386   figure(5);
387   plot(iss,ptol);
388   set(gca, 'XScale', 'log')
389   grid on
390   xlabel('Time steps','fontsize',12);
391   ylabel('Pressure drop ','fontsize',12);
392   figure(6);
393   plot(iss,cc);
394   set(gca, 'XScale', 'log')
395   grid on
396   xlabel('Time steps','fontsize',12);
397   ylabel('Pressure constant c','fontsize',12);
398
399   figure(7);
400   scatter(hy/ly,error);
401   set(gca, 'YScale', 'log')
402   set(gca, 'XScale', 'log')
403   b = polyfit(log(hy/ly), log(error'), 1);
404   wetfit = exp(b(2)) .* (hy/ly).^b(1);
405   hold on
406   ph1 = plot(hy/ly, wetfit,'r');
```

```
407   hold off
408   hold on
409   ph3=refline(2);
410   legend( [ph1 ph3],{sprintf('Slope = %f',b(1)),'Slope = 2' },...
411   'Location', 'NorthEast' );
412   hold off;
413   grid on;
414   xlabel('hy/D','fontsize',12);
415   ylabel('Inlet Velocity L2-Norm','fontsize',12);
416
417   figure(8);
418   slope7=(log(errorp(nh-1))-log(errorp(2)))/(log(hy(nh-1))-log(hy(2)))
419   scatter(hy/ly,errorp);
420   set(gca, 'YScale', 'log')
421   set(gca, 'XScale', 'log')
422   b = polyfit(log(hy/ly), log(errorp'), 1);
423   wetfit = exp(b(2)) .* (hy/ly).^b(1);
424   hold on
425   ph1 = plot(hy/ly, wetfit,'r');
426   hold off
427   hold on
428   ph3=refline(2);
429   legend( [ph1 ph3],{sprintf('Slope = %f',b(1)),'Slope = 2'},...
430   'Location', 'NorthEast' );
431   hold off;
432   grid on;
433   xlabel('hy/D','fontsize',12);
434   ylabel('Pressure Gradient L2-Norm','fontsize',12);
435
436  %%%%%%% PRINT OUT %%%%%
437  fprintf(['1) Theoretical flow rate for \n a parabolic velocity profile'...
438      ' initially: = %d [m^3/s] \n'], Q);
439  fprintf('2) Calculated flow rate initially: Ru* = %d [m^3/s] \n', qvc);
440  fprintf('3) The imposed volumetric flow rate: Q = %d [m^3/s] \n', qv);
441  fprintf('4) The volumetric flow rate finally: Ru* = %d [m^3/s] \n \n', R*ut);
442
443  fprintf(['A) Max. velocity calculated by Hagen-Poisouille Eq. for ...
         initial\n'...
444  'flow rate %d [m^3/s] is = %d [m/s] \n'],Q,vmax);
445  fprintf(['B) Max. velocity calculated by Hagen-Poisouille Eq. for final\n'...
446      'flow rate %d [m^3/s] is = %d [m/s] \n'],qv,vmax2);
447  fprintf(['C) Max. velocity after %d iterations and total time %d [s] =\n'...
448      '%d [m/s] \n\n '], is,is*dt, vmax1);
449  fprintf(['Initial theoretical pressure gradient calculated after %d '...
450      'iterations \n by Hagen-Poisouille Eq. = %d \n'], is, dpdx1);
451  fprintf(['Final theoretical pressure gradient calculated after %d '...
452      'iterations \n by Hagen-Poisouille Eq. = %d \n'], is, dpdx2);
453  fprintf('Slope calculated numerically = %d \n\n', slop/lxx);
454  fprintf('||p^(n+1)-p^(n)|| = %d \n\n', ((sum((pf).^2))^0.5));
455  fprintf('The value for c = %d \n', c);
456  fprintf('|c^(n+1)-c^(n)| = %d \n\n', abs(cf));
457  Ru=pf3*dt/qv
458  div=sum(pf1)
459  Re=rho*vmax2*ly/b
460  cfl=vmax2*dt/hx
```

Appendix D

Dynamic Smagorinsky model

The grid filtering with homogeneous isotropic filters:

$$\overline{U}(x.t) := \int U(x-r,t)G(|r|;\overline{\Delta})dr, \tag{D.1}$$

where the filter width, Δ is proportional to the grid spacing h, and the equations are solved for $\overline{U}$. The test filter $\widetilde{\Delta}$, which is usually equal to $2\overline{\Delta}$ can be expressed as:

$$\widetilde{U}(x.t) := \int U(x-r,t)G(|r|;\widetilde{\Delta})dr, \tag{D.2}$$

where $\widetilde{U}$ is unknown in the LES. Thus, a doubly filtered velocity field is defined as

$$\begin{aligned}\widetilde{\overline{U}}(x,t) &:= \int \overline{U}(x-r,t)G(|r|;\widetilde{\Delta})dr \\ &= \int U(x-r,t)G(|r|;\widetilde{\overline{\Delta}})dr.\end{aligned} \tag{D.3}$$

The effective double filter is

$$\widetilde{\overline{\Delta}} = \begin{cases} \widetilde{\Delta}, & \text{for the sharp spectral filter} \\ \left(\widetilde{\Delta}^2 + \overline{\Delta}^2\right)^{1/2}, & \text{for the Gaussian filter.} \end{cases} \tag{D.4}$$

By adding and subtracting the doubly filtered velocity, a decomposition of the velocity is obtained of the form

$$U = \widetilde{\overline{U}} + \left(\overline{U} - \widetilde{\overline{U}}\right) + u^{'}. \tag{D.5}$$

The smallest resolved motions $\overline{U} - \widetilde{\overline{U}}$ (eddies of size between $\overline{\Delta}$ and $\widetilde{\overline{\Delta}}$) can be found by using $\overline{U}$. They are resolved by using the grid filter $\overline{\Delta}$, but they are not captured with the test filter.

Now by using Germano's identity the residual stresses can be defined with the single and double filtering operations separately:

$$\tau_{ij}^R := \overline{U_i U_j} - \overline{U}_i\,\overline{U}_j \quad \text{(grid filter)}, \tag{D.6}$$

$$T_{ij} := \widetilde{\overline{U_i U_j}} - \widetilde{\overline{U}}_i\,\widetilde{\overline{U}}_j \quad \text{(double filter)}. \tag{D.7}$$

The terms $\overline{U_i U_j}$ and $\widetilde{\overline{U_i U_j}}$ cannot be computed from the LES model. Thus, by applying the test filter to Eq. (D.6) and subtracting it from the double filter residual stress Eq. (D.7), the resolved stress equation can be obtained, which is possible to be computed from the LES model.

$$\mathcal{L}_{ij} := T_{ij} - \widetilde{\tau_{ij}^R} = \widetilde{\overline{U}_i \, \overline{U}_j} - \widetilde{\overline{U}}_i \, \widetilde{\overline{U}}_j. \tag{D.8}$$

The resolved stress $\mathcal{L}_{ij}$ contributes to the residual stress from the largest unresolved motions. The Smagorinsky model for the anisotropic part is

$$\tau_{ij}^r := \tau_{ij}^R - \frac{1}{3}\tau_{kk}^R \delta_{ij} = -2c_S \overline{\Delta}^2 \overline{\mathcal{S}} \, \overline{S}_{ij}. \tag{D.9}$$

By using the same model for the filter width of $\widetilde{\overline{\Delta}}$ (double filter) the following expression is obtained

$$T_{ij}^a := T_{ij} - \frac{1}{3} T_{kk} \delta_{ij} = -2c_S \widetilde{\overline{\Delta}}^2 \, \widetilde{\overline{\mathcal{S}}} \, \widetilde{\overline{S}}_{ij}. \tag{D.10}$$

$M_{ij} := 2\overline{\Delta}^2 \widetilde{\overline{\mathcal{S}} \, \overline{S}}_{ij} - 2\widetilde{\overline{\Delta}}^2 \, \widetilde{\overline{\mathcal{S}}} \, \widetilde{\overline{S}}_{ij}$ is defined for simplifications assuming c_S is uniform. Furthermore, the Smagorinsky model of the anisotropic part of the resolved stress is:

$$\mathcal{L}_{ij}^S := T_{ij}^a - \widetilde{\tau_{ij}^r} = c_S M_{ij}, \tag{D.11}$$

where the anisotropic part of $\mathcal{L}_{ij}$ is

$$\mathcal{L}_{ij}^a := \mathcal{L}_{ij} - \frac{1}{3}\mathcal{L}_{kk}\delta_{ij}. \tag{D.12}$$

Now, the optimal value for c_S should be selected, which provides the best approximation. It is important to note that $\mathcal{L}_{ij}^a$ and $\mathcal{L}_{ij}^S$ have five independent components. Therefore, mean-square error minimization is used in order to find a suitable c_S value that satisfies all the components:

$$c_S = \frac{M_{ij}\mathcal{L}_{ij}}{M_{kl}M_{kl}}. \tag{D.13}$$

In channel flow, the value of c_S that is obtained by using Eq. (D.13) leads to fluctuations because of the low correlation between the rate of strain and the stress. Therefore, another method can be used to obtain stable results for LES. With this method, the value for c_S is obtained by taking averages:

$$c_S = \frac{(M_{ij}\mathcal{L}_{ij})_{avg}}{(M_{kl}M_{kl})_{avg}}, \tag{D.14}$$

and this method yields good results for transitional and fully turbulent channel flow [2]. It also provides the right value at the wall and for the laminar flow. Moreover, there is no need for additional wall treatment if the grid is fine enough to resolve the near-wall region.

Appendix E

Influence of the interface on turbulence

The increase in the velocity of the gas phase can produce waves at the interface. The wavy interface can be modeled as a solid surface with appropriate an surface roughness, in which, the liquid phase acts as a stationary wall since it has a smaller velocity than the gas phase. The velocity difference between two phases creates a shear stress, which can be called as *interfacial shear stress*.

In addition to the interfacial shear stress, the wall shear stress also needs to be calculated in order to model the momentum transfer between phases. However, the wall stress at the wall is different for both liquid and gas phases. Therefore, first, a single-phase wall shear stress is calculated by assuming only the liquid or gas phase flows in the pipe. Then, in a later stage, the two-phase wall shear stress is calculated by applying a multiplier to the single-phase shear stress. This method is called a *separated flow* model, which differs from a *homogeneous-flow* model in the sense that both phases are not assumed to flow together at the same velocity. Also, in homogeneous flow model, the two phases are assumed to flow as a single-phase with weighted averaged properties of both phases. On the other hand, the *two-fluid* model is the most complicated and detailed model, which requires information about flow properties of both phases separately, and accurate models for the transfer of mass, momentum, and energy between the phases at the interface.

The interface needs to be considered as a material surface when there are surfactants, which means it has its own density and momentum. However, in this study, the density of the interface is assumed to be zero. Hence, according to the jump conditions, mass and momentum balance at the interface (i.e., Eqs. (3.5) and (3.15)) can be applied. For the most general case, the mass transfer of the kth phase at the interface is defined by

$$\dot{m^k} = \rho^k (u^k - w) \cdot n^k. \tag{E.1}$$

The mass balance for all phases is

$$\sum_k \dot{m}^k = \sum_k (\rho^k (u^k - w) \cdot n^k) = 0. \tag{E.2}$$

The balance of momentum for the kth phase throughout the interface (i.e., Eq. (3.15)) yields:

$$\sum_k \left(u^k \dot{m}^k - n^k \cdot (\tau^k - p^k \mathbb{1})\right) = \sigma n^k \nabla_s \cdot n^k - \nabla_s \sigma, \tag{E.3}$$

where ∇_s is the surface gradient operator.

$$\tau^k = \mu^k S_D^k \tag{E.4}$$

is the deviatoric stress tensor, S_D^k is defined as

$$S_D^k = \left(S^k - \frac{1}{3}\mathrm{tr}(S^k)\mathbb{1}\right), \tag{E.5}$$

where $S = \nabla u + \nabla^T u$ is the deformation rate tensor, and tr stands for the trace of a tensor.

In order to extend each phase to the whole domain, there is a need to define the interface of the kth phase as a function of position and time. For each phase a *phase indicator function or characteristic function* $\chi^k(\mathrm{x}, t)$ can be defined as:

$$\chi^k(x,t) = \begin{cases} 1, & \text{if x} \in \text{ the } k\text{th phase} \\ 0, & \text{if x} \notin \text{ the } k\text{th phase} \end{cases} \tag{E.6}$$

If the given position vector is in the interface (i.e., x $\in S$), then the function $\chi^k = 0$, the other way round is also true. The function χ^k is defined such that it complies several properties. The two phase indicator functions are related:

$$\chi^1 + \chi^2 = 1. \tag{E.7}$$

The gradient of the phase indicator function can be expressed in terms of the Dirac delta function of the interface $\delta(\mathrm{x}-\mathrm{x}_i, t)$, where x_i is a point on the interface, and the interface unit normal vector n^k:

$$\nabla \chi^k = n^k \delta(\mathrm{x} - \mathrm{x}_i, t). \tag{E.8}$$

Moreover, the material derivative of χ^k is zero, which follows from the topological equation:

$$\frac{\partial \chi^k}{\partial t} + w \cdot \nabla \chi^k = 0. \tag{E.9}$$

The conservation of mass equation can be augmented by using a phase indicator function respectively:

$$\frac{\partial \chi^k \rho^k}{\partial t} + \nabla \cdot (\chi^k \rho^k u^k) = \rho^k (w - u^k) \cdot n^k \delta_i, \tag{E.10}$$

where $\delta_i = -n^k \cdot \nabla \chi^k$ is the Dirac function centered on the interface in order to describe the behavior of each phases near the interface. By applying the phase indicator function to the balance of momentum equation at the interface yields

$$\frac{\partial \chi^k \rho^k u_i}{\partial t} + u_j \frac{\partial \chi^k \rho^k u_i}{\partial x_j} = -\frac{\partial \chi^k p}{\partial x_i} + \frac{\partial}{\partial x_j}\left[\chi^k \mu_{eff}\left(\frac{\partial \overline{u}_i}{\partial x_j} + \frac{\partial \overline{u}_j}{\partial x_i}\right)\right] - \chi^k \rho^k g_i. \tag{E.11}$$

The mass and momentum conservation equations defined by Eqs. (E.10) and (E.11) need to be modeled with respect to thermodynamic properties.

An approach called *one-fluid* modeling, where a single set of equations is solved for all the phases of the flow, can be used for multiphase flows by defining exactly one-fluid variable as the sum of the phase indicator weighted phases:

$$\phi = \sum_k \chi^k \phi^k. \tag{E.12}$$

One-fluid mass and momentum conservation equations can be written by summing all the phases of the flow. This way, the equation for the mass conservation can be simplified by using the mass jump condition, i.e., Eq. (E.2):

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho u) = 0. \tag{E.13}$$

The momentum conservation of the one-fluid approach (by summing all the phases) is given by

$$\frac{\partial \rho u_i}{\partial t} + u_j \frac{\partial \rho u_i}{\partial x_j} = -\frac{\partial p}{\partial x_i} + \frac{\partial}{\partial x_j}\left[\mu_{eff}\left(\frac{\partial \overline{u}_i}{\partial x_j} + \frac{\partial \overline{u}_j}{\partial x_i}\right)\right] - \rho g_i. \tag{E.14}$$

This equation can be simplified using the jump condition for the momentum, i.e., Eq. (E.3):

$$\begin{aligned}\frac{\partial \rho u_i}{\partial t} + u_j \frac{\partial \rho u_i}{\partial x_j} &= -\frac{\partial p}{\partial x_i} + \frac{\partial}{\partial x_j}\left[\mu_{eff}\left(\frac{\partial \overline{u}_i}{\partial x_j} + \frac{\partial \overline{u}_j}{\partial x_i}\right)\right] - \rho g_i \\ &= \nabla_s \cdot n^k \sigma n^k \delta_i - \nabla_s \sigma \delta_i.\end{aligned} \tag{E.15}$$

The above mentioned one-fluid approach has been introduced by [26] in order to obtain results for multiphase flows by using classical single-phase flow models.

E.1 The baseline method: Mass-Conserving evel Set (MCLS) method

E.1.1 Interface model with the MCLS method

The starting point of this thesis is a specific version of the MCLS method developed for discretization of the equations that describe immiscible incompressible two-phase flow in a circular pipe geometry. There exists complex interface topologies in between two fluids. This interface is a moving (internal) boundary. The interface can be modeled explicitly (moving mesh) or implicitly (fixed mesh) or as a combination of both. It is elaborate to simulate large numbers of different interfaces with moving meshes. Moving boundary problems can be solved with two approaches: *interface tracking* and *interface capturing*. In interface tracking, the moving boundary (interface) is described by meshes. On the other hand, in interface capturing the interface is described implicitly by an artificial scalar field.

Interface tracking methods are not applicable when there is a sharp change in the topology since the marked interface is tracked from the initial scheme, which is assumed to be not changing throughout the simulation. It is more difficult to compute the interface when

it has arbitrary shape and topology. For this reason, in this study an interface capturing method is considered, which is a combination of *Level Set (LS)* and *Volume of Fluid (VOF)* methods.

In the LS method, the interface can be expressed at a given time t as the zero LS of a function called the LS function $\Phi(x,t)$. By that means, the initial surface is defined as $\{x|\ \Phi(x,0)=0\}$, $\Phi>0$ inside fluid 1 and $\Phi<0$ otherwise. The interface is shifting by the advection of Φ as a material property:

$$\frac{\partial \Phi}{\partial t} + u \cdot \nabla \Phi = 0. \tag{E.16}$$

The main problem with this method is that the material volume is not conserved well.

VOF defines a function, which defines the volume of a computational grid cell, and tracks the grid cells. Thereby, the fraction of each fluid phase in a grid cell is known. A marker function Ψ gives the fractional volume of specified fluid in a computational cell. For example, in a grid cell Ω, Ψ is defined by

$$\Psi = \frac{1}{vol(\Omega)} \int_{\Omega} \chi d\Omega. \tag{E.17}$$

where χ is the characteristic function, which has value 1 in fluid 1 and 0 elsewhere. The value of Ψ changes very fast throughout the interface because of the definition of the characteristic function. Therefore, it is difficult to compute Ψ after each transport step of the interface. Although, VOF methods are conserving mass, they are not the best choice for our problem. The reason for that is the difficulty of computing interface derivatives due to the jumps (step-like behavior) in the marker function.

Interface tracking methods mainly deal with the dynamic behavior of the interface. In addition to these methods, there exists a technique called the reconstruction of material interfaces. This method adds discrete pieces or piecewise functions of an interface together to rebuild the continuous interface. Examples for material reconstruction are *simple line interface (SLIC)* and *piecewise linear interface construction (PLIC)*. Although, these methods are very accurate, these approaches are not considered since interface reconstruction is a difficult process.

The flow-field can be solved easily with the LS method because of the relation between the LS function Φ and interface normals, curvature and distance to the interface. However, mass is not conserved in the LS method when the interface advects. In order to conserve mass, the VOF function Ψ is used together with the LS function Φ explicitly, and this is called the *Mass-Conserving Level Set (MCLS)* method [27]. The interface is assumed to be piecewise linear within a computational cell, and it is defined as:

$$\Psi = f(\Phi, \nabla\Phi). \tag{E.18}$$

By using Eq. (E.18), the advection of the interface can be computed with less effort, and Φ can be obtained directly from Ψ. The smoothing operation of Ψ is not necessary when using MCLS method. Moreover, in three-dimensional space the implementation of MCLS method is much easier than VOF method [27].

In MCLS, in order to conserve mass up to a specified tolerance, the LS function is corrected by using the fraction of Ψ in that computational cell. The fractional volume of a fluid in a given computational cell is calculated by using the LS function Φ^n; $\Psi = f(\Phi, \nabla\Phi)$.

For the first step, the LS function is advected, and re-initialization is carried out in order to smooth the function Φ. Next, Ψ evolves in time, while conserving mass, and this advected VOF function can be called as Ψ^{n+1}. Then, by using Ψ^{n+1}, Level-Set function at new time step $\Phi^{n+1,*}$ is corrected in order to find $\Psi^{n+1} = f(\Phi^{n+1}, \nabla\Phi^{n+1})$, which holds for this new Level-Set function and conserves mass [27].

E.1.2 Navier-Stokes mimetic discretization

The continuity, momentum, and energy equations are the mathematical definitions of three fundamental physical processes, and computational fluid dynamics is based on three physical principles, which are obtained by these governing equations of fluid dynamics:

- Mass is conserved,
- Momentum is conserved,
- Energy is conserved.

Model for a specific flow problem can be established by using the appropriate fundamental physical principles from above. From this model, necessary mathematical equations are defined, which express these physical principles. Therefore, our discretization technique should satisfy these principles to be able to discretize the mathematical model without loss of accuracy.

The discretization based on mimetic finite discrete first order operators (divergence, gradient, curl) mimics the properties of the original continuum differential operators; e.g., conservation laws, vector and tensor calculus, and fundamental identities. The Navier-Stokes equations in cylindrical coordinates possess some difficulties such as the coordinate singularity that occurs at the axis $r = 0$. In the mimetic approach, the problem near the axis is vanished by defining these discrete operators such that they mimic the properties of vector analysis in continuum case (i.e. $curl(grad) = 0$, and $div(curl) = 0$). More specifically, the discrete divergence and gradient operators are negative adjoint of each other with respect to the standard L^2 inner products. The choice of cylindrical coordinates is due to the fact that the computations become easier due to the orthogonality of the grids [28].

The properties of discrete operators in Cartesian coordinates can be analyzed, and can be realized that they also have similar definitions as already defined above in mimetic approach. For example, in Cartesian coordinates, negative of the divergence operator **-D** is the transpose of the gradient operator **G** with respect to suitable inner product if and only if particular boundary condition is satisfied. **D** and **G** operators are defined by

$$\begin{aligned} \mathbf{D} &: u \rightarrow \frac{1}{\rho}\nabla \cdot u, \\ \mathbf{G} &: \phi \rightarrow \frac{1}{\rho}\nabla\phi, \end{aligned} \tag{E.19}$$

where ρ is the density. Inner products of these operators yields a boundary condition as follows:

$$\int_{\partial\Omega} \phi u \cdot dS. \tag{E.20}$$

For velocity and pressure, the boundary condition can be written as:

$$\int_{\partial\Omega} p u \cdot dS. \tag{E.21}$$

For Dirichlet boundary condition, $p = 0$ at the boundary, and the integral becomes zero. For Neumann boundary condition, $u \cdot dS = u \cdot n \; dA = 0$, so the integral vanishes again.

In our approach, the discretization is carried out by using a finite difference method, which solves the incompressible Navier-Stokes equations in a cylindrical coordinate system. It is more difficult to use cylindrical coordinates, when the domain has a coordinate singularity, i.e., $r = 0$. Nevertheless, it is possible to overcome this problem by using available approaches in the literature. From those approaches, The mimetic approach is chosen for this thesis because it has some important advantages. For example, there is no need to approximate any additional term near the axis, and it is applicable everywhere in the domain [28].

Discrete spatial operators are defined such that they preserve most of the properties of the continuum operators. Mimetic finite difference operators are well-suited for this. However, they need a proper definition for inner products and discrete spaces for a proper use.

The components of discrete vector fields $\underline{\mathbf{U}}$ are defined as normal to the centers of the cell faces in the space HS. The components of $\underline{\mathbf{U}}$ are not defined at $r = 0$, and they are defined with zero normal component on the boundary. The space HS is defined as follows:

$$\mathrm{HS} = \left\{ \underline{\mathbf{U}} = \left(u_{i+1,j+1/2,k+1/2}, v_{i+1/2,j,k+1/2}, w_{i+1/2,j+1/2,k} \right) \right\}. \tag{E.22}$$

The inner product definition for the space HS is

$$\begin{aligned}\left(\underline{\mathbf{U}}, \underline{\mathbf{U}}^{'}\right)_{\mathrm{HS}} = & \sum_{i=0}^{i=I-1} \sum_{j=0}^{j=J-1} \sum_{k=0}^{k=K-1} \frac{\Delta r \Delta\theta \Delta z}{2} \\ & [r_{i+1} u_{i+1,j+1/2,k+1/2} u^{'}_{i+1,j+1/2,k+1/2} + r_i u_{i,j+1/2,k+1/2} u^{'}_{i,j+1/2,k+1/2} \\ & + r_{i+1/2}(v_{i+1/2,j+1,k+1/2} v^{'}_{i+1/2,j+1,k+1/2} + v_{i+1/2,j,k+1/2} v^{'}_{i+1/2,j,k+1/2} \\ & + w_{i+1/2,j+1/2,k+1} w^{'}_{i+1/2,j+1/2,k+1} + w_{i+1/2,j+1/2,k} w^{'}_{i+1/2,j+1/2,k})].\end{aligned} \tag{E.23}$$

where $\Delta r = 1/I$, $\Delta\theta = 2\pi/J$, and $\Delta z = H/K$ are the increments of staggered uniform grid in the discretized computational domain. The discrete scalar function $\boldsymbol{\phi}$ is defined at the cell centers in the space HC:

$$\mathrm{HC} = \boldsymbol{\phi} = (\phi_{i+1/2,j+1/2,k+1/2}). \tag{E.24}$$

The space HC has the following inner product:

$$\left(\boldsymbol{\phi}, \boldsymbol{\phi}^{'}\right)_{\mathrm{HC}} = \sum_{i=1}^{i=I} \sum_{j=1}^{j=J} \sum_{k=1}^{k=K} \phi_{i-1/2,j-1/2,k-1/2} \phi^{'}_{i-1/2,j-1/2,k-1/2} r_{i-1/2} \Delta r \Delta\theta \Delta z. \tag{E.25}$$

The discrete vector fields $\underline{\mathbf{V}}$ is defined at the mid edges of the cells in the space HL:

$$\mathrm{HL} = \{\mathbf{V} = (\eta_{i+1/2,j,k}, \omega_{i,j+1/2,k}, \zeta_{i,j,k+1/2})\}. \tag{E.26}$$

The discrete scalar function $\boldsymbol{\psi}$ is defined at the vertices (i, j, k) of the cells in the space HN:

$$\mathrm{HN} = \{\boldsymbol{\psi} = (\psi_{i,j,k})\}. \tag{E.27}$$

The gradient operator is defined as the negative adjoint of the discrete divergence operator with respect to the given inner products:

$$(\boldsymbol{\phi}, \overline{\mathbf{D}}_S(\underline{\mathbf{U}}))_{\mathrm{HC}} = -(\overline{\mathrm{G}}(\boldsymbol{\phi}, \overline{\mathbf{U}})_{\mathrm{HS}}, \tag{E.28}$$

where the gradient operator $\overline{\mathrm{G}}$ maps HC onto HS, and the divergence operator $\overline{\mathbf{D}}_S$ maps HS onto HC. The curl operator $\overline{\mathbf{C}}_S$ maps HS onto HL, where discrete vorticity components are defined at the edge centers [28]. The usual properties of vector analysis such as $curl(grad) = 0$ and $div(curl) = 0$ can be checked with the given discrete operators;

$$\overline{\mathbf{C}}_S(\overline{\mathrm{G}}(\boldsymbol{\phi})) = 0 \quad \forall \boldsymbol{\phi} \text{ in HC}, \tag{E.29}$$

$$\overline{\mathbf{D}}_L(\overline{\mathbf{C}}_S(\underline{\mathbf{U}})) = 0 \quad \forall \underline{\mathbf{U}} \text{ in HS}, \tag{E.30}$$

where $\overline{\mathbf{D}}_L$ is the divergence operator, which is a mapping from HL onto HN.

Artificial sources of energy and vorticity may lead to unphysical long terms in LES. Above all, mimetic schemes are useful for LES because of vorticity preserving discretizations. When a mimetic curl operator is applied to the discrete momentum equation, a consistent discrete vorticity equation is obtained, which is called the vorticity preservation [29].

Bibliography

[1] J. G. M. Eggels, *Direct and large eddy simulation of turbulent flow in a cylindrical pipe geometry* (TU Delft, Delft University of Technology, 1994).

[2] S. B. Pope, *Turbulent flows* (IOP Publishing, 2001).

[3] C. Vallée, T. Höhne, H.-M. Prasser, and T. Sühnel, *Experimental investigation and CFD simulation of horizontal stratified two-phase flow phenomena,* Nuclear Engineering and Design **238**, 637 (2008).

[4] S. Ghorai and K. Nigam, *CFD modeling of flow profiles and interfacial phenomena in two-phase flow in pipes,* Chemical Engineering and Processing: Process Intensification **45**, 55 (2006).

[5] D. C. Wilcox *et al.*, *Turbulence modeling for CFD*, Vol. 2 (DCW industries La Canada, CA, 1998).

[6] M. Birvalski, *Experiments in stratified gas-liquid pipe flow* (TU Delft, Delft University of Technology, 2015).

[7] M. Avila, *Lecture notes in Reynolds averaged Navier-Stokes (RANS),* (2015).

[8] R. Absi, *A simple eddy viscosity formulation for turbulent boundary layers near smooth walls,* Comptes Rendus Mecanique **337**, 158 (2009).

[9] P. Gnambode, P. Orlandi, M. Ould-Rouiss, and X. Nicolas, *Large-eddy simulation of turbulent pipe flow of power-law fluids,* International Journal of Heat and Fluid Flow **54**, 196 (2015).

[10] M. Wörner, *A compact introduction to the numerical modeling of multiphase flows* (Forschungszentrum Karlsruhe, 2003).

[11] J. Eggels, F. Unger, M. Weiss, J. Westerweel, R. Adrian, R. Friedrich, and F. Nieuwstadt, *Fully developed turbulent pipe flow: a comparison between direct numerical simulation and experiment,* Journal of Fluid Mechanics **268**, 175 (1994).

[12] J. Eggels and F. Nieuwstadt, *Large eddy simulation of turbulent flow in an axially rotating pipe,* in *Proceeding of the 9th Symposium on Turbulent Shear flows* (1993) pp. 310–1.

[13] B. Boersma and F. Nieuwstadt, *Large-eddy simulation of turbulent flow in a curved pipe,* Journal of Fluids Engineering **118**, 248 (1996).

[14] G. Chinello, *CFD modelling of stratified two-phase flow for oil and gas transportation*, Master's thesis, University of Pisa (2015).

[15] P. Lombardi, V. De Angelis, and S. Banerjee, *Direct numerical simulation of near-interface turbulence in coupled gas-liquid flow,* Physics of Fluids (1994-present) **8**, 1643 (1996).

[16] M. Fulgosi, D. Lakehal, S. Banerjee, and V. De Angelis, *Direct numerical simulation of turbulence in a sheared air–water flow with a deformable interface,* Journal of Fluid Mechanics **482**, 319 (2003).

[17] E. Labourasse, D. Lacanette, A. Toutant, P. Lubin, S. Vincent, O. Lebaigue, J.-P. Caltagirone, and P. Sagaut, *Towards large eddy simulation of isothermal two-phase flows: Governing equations and a priori tests,* International Journal of Multiphase Flow **33**, 1 (2007).

[18] N. Lamarque, B. Zoppé, O. Lebaigue, Y. Dolias, M. Bertrand, and F. Ducros, *Large-eddy simulation of the turbulent free-surface flow in an unbaffled stirred tank reactor,* Chemical Engineering Science **65**, 4307 (2010).

[19] P. A. Berthelsen and T. Ytrehus, *Calculations of stratified wavy two-phase flow in pipes,* International Journal of Multiphase Flow **31**, 571 (2005).

[20] A. Ullmann and N. Brauner, *Closure relations for two-fluid models for two-phase stratified smooth and stratified wavy flows,* International Journal of Multiphase Flow **32**, 82 (2006).

[21] T. Frank, *Numerical simulations of multiphase flows using CFX-5,* in *CFX Users conference, Garmisch-Partenkirchen, Germany* (2003).

[22] L. M. Lakehal, D and C. Narayanan, *Advances in the large–eddy and interface simulation (leis) of interfacial multiphase flows in pipes,* Progress in Computational Fluid Dynamics, an International Journal **12**, 153 (2012).

[23] D. Lakehal, M. Laboisa, D. Caviezela, and B. Belhouachib, *Transition of gas-liquid stratified flow in oil transport pipes,* J. Eng. Res **8**, 49 (2011).

[24] J. Van Kan, *A second-order accurate pressure-correction scheme for viscous incompressible flow,* SIAM Journal on Scientific and Statistical Computing **7**, 870 (1986).

[25] G. Segal, K. Vuik, and K. Kassels, *On the implementation of symmetric and anti-symmetric periodic boundary conditions for incompressible flow,* International journal for numerical methods in fluids **18**, 1153 (1994).

[26] I. Kataoka, *Local instant formulation of two-phase flow,* International Journal of Multiphase Flow **12**, 745 (1986).

[27] S. Van der Pijl, A. Segal, C. Vuik, and P. Wesseling, *A mass-conserving level-set method for modelling of multi-phase flows,* International Journal for Numerical Methods in Fluids **47**, 339 (2005).

[28] E. Barbosa and O. Daube, *A finite difference method for 3D incompressible flows in cylindrical coordinates,* Computers & Fluids **34**, 950 (2005).

[29] A. Abba and L. Bonaventura, *A mimetic finite difference discretization for the incompressible navier-stokes equations,* International Journal for Numerical Methods in Fluids **56**, 1101 (2008).