UNIVERSITY OF THESSALY

SCHOOL OF ENGINEERING

DEPARTMENT OF MECHANICAL ENGINEERING

Numerical Study of Single-Phase Impinging Jet Flows

by

IOSIF LEVENTAKIS

Submitted in partial fulfillment of the requirements for the degree of Diploma in Mechanical Engineering at the University of Thessaly

Volos, 2023

Approved by the Committee on Final Examination:

Supervisor Dr. Konstantinos Ritos,

Professor, Department of Mechanical Engineering, University of Thessaly

Member Dr. Georgios Charalampous,

Professor, Department of Mechanical Engineering, University of Thessaly

Member Dr. Nikolaos Pelekasis,

Professor, Department of Mechanical Engineering, University of Thessaly

Date Approved: 16 February, 2023

ACKNOWLEDGEMENT

Before moving on with the presentation of the thesis, I would like to express my deep gratitude to my supervisor Dr. Konstantinos Ritos for his support and his continuous guidance throughout this project. Apart from the countless hours of meetings through whom he shared his knowledge with me, he also gave me the opportunity to get to know with the field of CFD, which is of great interest. For the first time in my bachelor years, I focused that much on an engineering field, which meets its practice in various industries, and as a result it is going to be a great asset for me and my future as an engineer. I would definitely not have had this opportunity if it wasn’t for him. Lastly, I would like to thank my mother for her assistance with some parts of the English language used in the thesis.

ABSTRACT

Impinging jets are widely used in industry as a method of achieving high rates of heat and mass transfer. Although their study focuses on heat transfer, because of the wide range of applications it involves, this thesis focused on the mass transfer aspects of these flows, and was written in order to shed light on their characteristics, since the literature review highlights the need for further investigation. This is because of their mostly unsteady nature and the wide range of flow phenomena that can be generated in such flows, such as large curvature involving strong shear and normal stresses, stagnation in the wall boundary layers, heat transfer with the impinged wall and small-scale turbulent mixing. Quality single-phase results can be used to predict multi-phase impinging jet flows and their erosion effects, a topic that lacks investigation and concerns various industries. The flow was turbulent in all cases examined, and special emphasis was given on the behaviour of the fluid near the impingement surface, where steep gradients exist. The investigation was carried out through numerical simulations of the fluid flow, in *OpenFOAM* software. Different parameters were examined throughout the thesis, via the validation and the verification of the results with the existing literature. The mesh, the 2D and 3D resolving of the flow, the y^+ value, the velocity, the wall shear stress, the Reynolds stresses and the kinematic viscosity of the fluid, were all part of the parameters examined. First, some 2D simulations were carried out with the goal of validating the results with experimental findings and checking the mesh dependency of the problem. Afterwards, two more 3D simulations were run with the aim of both validating and verificating the results. Then, for the same 3D simulations a parametric analysis was conducted, in order to test the effect of the flow's Reynolds number, through different bulk velocities and fluid kinematic viscosities. Finally, a reference to multi-phase impinging jets was made, in which their usefulness was highlighted, and extensive reference to possible future work was made.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

Unit 1. **LITERATURE REVIEW**

1.1 Jet impingement theory

1.1.1 Usefulness of jet impingement

The high heat and mass transfer rates of impinging jets are achieved thanks to the high turbulence and the thin thermal boundary layer that is formed in such high-rate fluid flows. Concerning heat transfer, the main advantage of this heating method is the easiness of producing a hot jet and aiming it at a specific region of a surface. Among other methods of convective heat transfer, such as natural convection (which is a result of density differences) or parallel forced flow by a blower, such jets have higher efficiency. Specifically, they can transfer or remove from a body up to 500 MW/m^2 of thermal energy (thermal power transfer density) [1]. However, in this thesis emphasis will be given on the mass transfer rates of these jets, since only properties such as velocity, shear stresses and Reynolds stresses are going to be studied. Single-phase impinging jet findings are crucial and make up the base before studying multi-phase impinging jets and their erosion effects, which are of great interest in industry.

1.1.2 Jet impingement physics

When the term «impinging jet» is used, automatically a jet of fluid collapsing onto a plate comes to mind. Whether the two fluids interacting (one coming out of the jet and a stationary one, where the jet expands) are of the same properties or not, jets can be divided into submerged (e.g., water in water) or unsubmerged (e.g., water in air). For both cases, three main regions are observed, although the free jet region may not actually exist if the distance between the nozzle and the plate of impingement is less than 2 nozzle diameters [2]. These regions are:

- The free jet region, originating from the nozzle exit up to a distance from the plate and which, in turn, consists of the potential core, the developing and the fully developed region.
- The stagnation zone which is close to the plate.
- The wall jet region which locates on either side of the stagnation zone [3].

For the case of the submerged impinging jet, which will be examined from now on and as shown in Figure 1, after the discharge from the nozzle, the flow enters the potential core region, in which an increasing thickness zone is observed. The geometric confines of this zone are defined from the nozzle exit to the point where the longitudinal mean velocity U reaches the value of 0.99U$_0$, where U$_0$ is the axial longitudinal velocity at the nozzle exit [4]. There, a «potential cone» is formed, inside which the flow has the same conditions with the nozzle’s exit conditions, in terms of velocity, pressure, temperature, density etc., while outside the cone, velocity decreases in order to reach the velocity of the environment. The above region follows the developing region, in which the axial velocity decays rapidly, due to large shear stresses at the boundary of the jet. Finally, the last part of the free jet region is where there is a fully developed velocity profile. Then, the stagnation/deflected zone follows, which is the region where the jet strikes onto the surface and the flow from axially decelerating becomes radially accelerating, resulting in an increase in static pressure. The last region is characterized by the complete deflection of the jet, strong shear forces and velocity fluctuations and it is the one that the flow evolves from stagnation to wall-jet, due to the jet finite width and the momentum exchange between the near-flow and the plate.

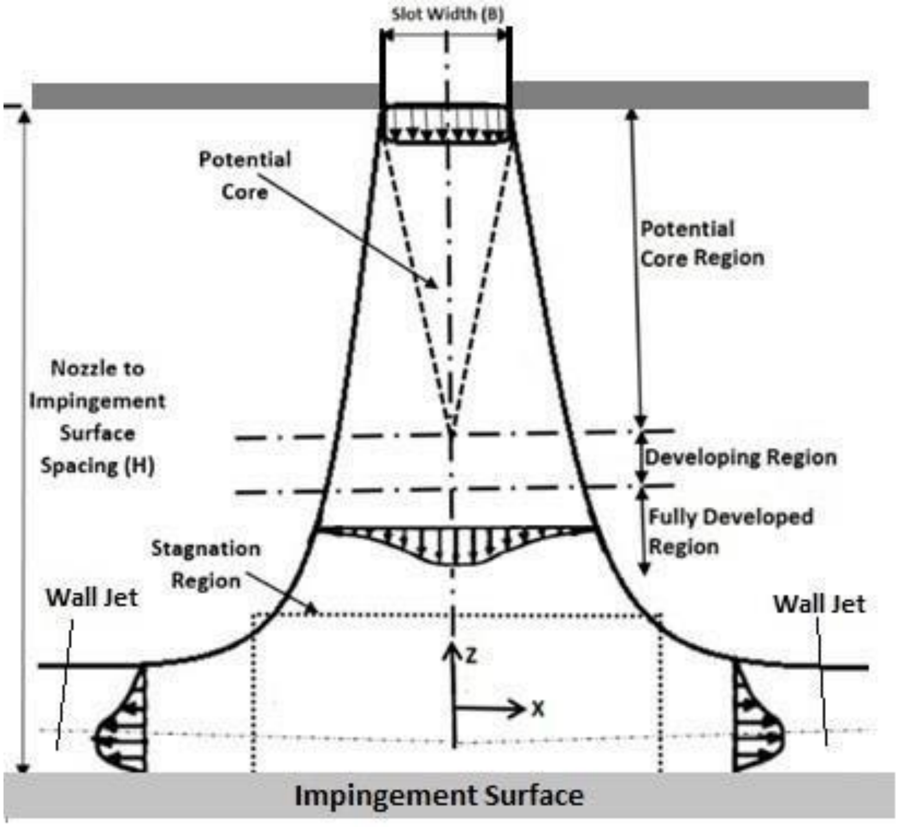


Figure 1. Different regions of a jet impingement flow [5].

1.1.3 Applications

As said earlier, impinging jets find implementation on multi-phase jets, where mass transfer is involved. By referring to the term «multi-phase», the simultaneous flow of materials with two or more thermodynamic phases is meant and, in our case, it is the interaction between liquid-solid or air-solid. Although there is limited research on two-phase, solid loaded impinging jets [6], nowadays more and more research is conducted, which is driven by the need to mitigate the erosion caused in wind turbines, slurry pumps etc. Erosion means damage, which results in less safety of the system, possible performance drop, as well as repairing and non-operation costs. In wind turbines, erosion appears when blades slice through air, which, as known, consists of particles of different diameters. Over time, these collisions wear off the turbine's surface, so repair is necessary. In pumps, erosion appears when sludge is transferred in large quantities at specific locations and in this case, damage can be much larger due to the system's bigger size. There, jet impingement occurs after the discharge to the surrounding structures, while in turbines, after the collision of the jet to the blade's surface.

Concerning heat transfer in which most applications of jet impingement exist, but will not be studied further in this thesis, by the start of the 20th century impinging jets had already been studied thoroughly and, since then, are used in a wide range of industrial applications which include heating, cooling and drying. Some of these applications, concerning single-phase impinging jets apply to turbine blade cooling, oil jet cooling technology, cooling of electronic devices, furnace heating and food processing. Specifically, in turbomachinery jet impingement cooling meets its practice via the cooling of the turbine's parts, including the guide vanes and the rotor disks and blades. This way, the lifetime of the components is increased, as they are protected from overheating. However, the method is challenging due to the complex geometry and the rotating parts of the turbine, as well as the turbulence of the flow [7]. Concerning the field of automotive, manufacturers use the method to prevent the overheating of the engine's pistons, by using oil as fluid. The need came in 1940 when pistons ran too hot and caused breakups and scoring of the cylinders. The way the cooling is achieved is through the supply of extra oil by a pump at the underside of the domes [8]. In the field of electronics, the increase of heat dissipation rate of the electronic systems to the value of 100 W/cm^2 has led to the searching of advanced cooling technologies. Unfortunately, despite the advantages of jet impinging cooling, in the specific field it is considered impractical for the moment. The reasons behind this have to do with the complex geometries of the devices, their high cost of manufacturing and the size of the pumping system [9]. In furnaces, flame jets on a surface are used to increase heat transfer and reduce fuel consumption and capital costs, despite their major drawback, which is the high non-uniformity of the heat flux [10]. Finally, as mentioned before, jet impinging technology is used in food-processing operations too, specifically for baking, drying, freezing and toasting of food. The innovation belongs to Smith who proposed the method for cooking in ovens in 1973. The method includes single or multiple air turbulent jets with Reynolds Number (Re) greater than 10,000 [11].

1.1.4 Research

The above applications are a result of thorough experimental and numerical investigation on impinging jets, although there are major drawbacks when using the experimental ones, which is consequence of the use of point measurement techniques, such as Hot Wire Anemometry (HWA) and Laser Doppler Anemometry (LDA) that are time consuming and unable to show the instantaneous evolution of the flow field [12]. Since Martin's pioneering review article (1973) on heat and mass transfer via impinging jets [13], many additional reviews have been published [4]. However, impinging jet phenomena are not yet fully understood, due to their unsteady nature and the great difficulties appearing while conducting both experimental and numerical investigations, so further study needs to be done [4]. Until now, it has been found that flow conditions and transfer rates in these jets depend on a number of factors, among which, the Reynolds number, the fluid Prandtl number, the nozzle exit geometry, the angle of impingement, the turbulence intensity and the nozzle-to-plate distance [12]. Below, the reader can follow some findings concerning some of these parameters, in single-phase flows.

Simionescu *et al.* [14] conducted a computational fluid dynamics (CFD) numerical study, regarding a circular laminar air jet impinging on a flat wall with smooth surface. For the simulation, the CFD package

FLUENT [15] developed by *Ansys® Academic Research Mechanical, Release 18.1* was used and the radial velocity profile at different radial positions and Reynolds numbers were studied. The positions studied were 0.5D, 1.5D, 3D and 6D, while the Reynolds numbers considered were 300, 420, 600 and 1,000. The results showed that at higher Reynolds numbers the maximum velocity appears closer to the wall, independent of the radial positions.

As far as the angles of impingement are concerned, those can occur by the sudden release of gasses from a storage medium, collapsing onto a surface. That made Adrian Kelsey *et al.* [16] numerically and experimentally investigate such flows. Two gasses were involved; nitrogen with a density of 4% lower than that of air and chlorodifluoromethane (R-22) with a density of three times more than that of air. For the experiment, a nozzle and a thermocouple were used and the angle of impingement was either 10° or 30° from the horizontal, with a distance of 0.50 m between the nozzle's exit and the impingement point. Gas impinged onto a smooth horizontal «table», measuring 4 m along the jet's axis and 3 m wide. During the experiment, velocity measurements were done using a thermal anemometer probe. For the CFD simulations, the domain increased horizontally to 5.5 m long and 5 m wide (cube), because boundaries affected the flow field and the commercial package used was CFX-5 developed by *ANSYS, Inc. (2015) ANSYS CFX-Solver Theory Guide, Release 16.2* [17]. Experimental results, as shown in figures 2, 3, showed that with angled impingement, maximum velocity decayed faster compared with axisymmetric free jets or wall jets. Also, it was found that the behaviour downstream the impingement point is something between that of a wall jet and normal impingement. Equivalent results were obtained by the EJECT integral model used in the numerical simulations. This model was developed for angled impingement of single-phase momentum jets, and is based on impingement angle dependent entrainment and lateral spreading rates.

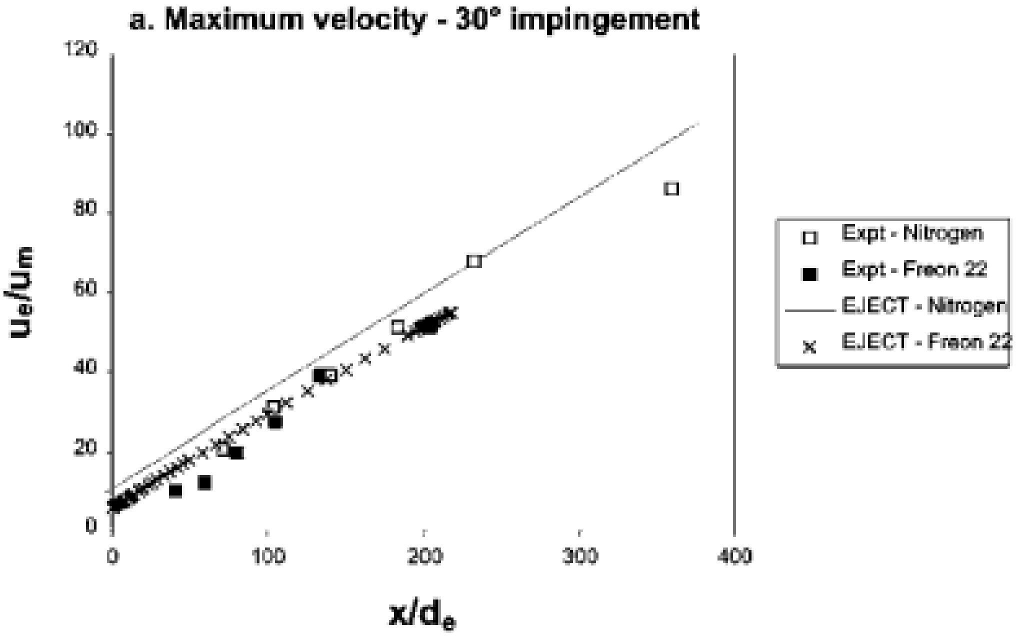


Figure 2. *Comparative experimental and CFD results for 30° angle of impingement [16].*

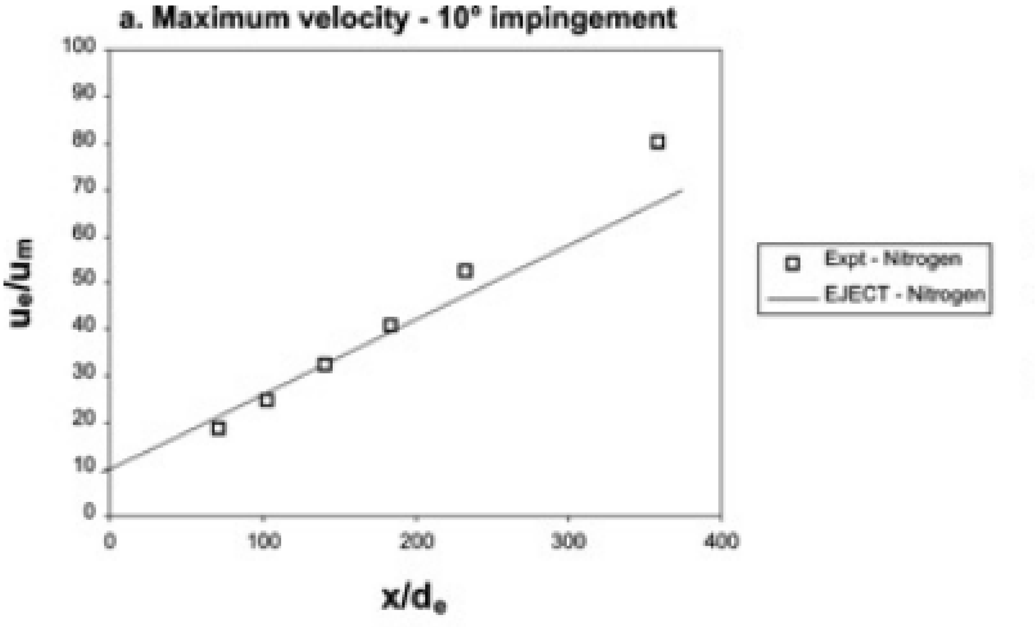


Figure 3. *Comparative experimental and CFD results for 10° angle of impingement [16].*

1.2 CFD and the Finite Volume Method

1.2.1 What is CFD?

As mentioned before, the above findings are a result of the branch of CFD, so now, a reference on it needs to be done. Specifically, CFD is a branch of fluid mechanics, which uses the fields of applied mathematics, physics and computer software in order to visualize the flows of fluids, with the use of computers for the calculation of the flow properties (e.g., velocity, pressure, temperature, density, viscosity). The need for its development came by the inability of mathematicians and engineers to analytically solve the partial differential equations encountered in flows. Historically, it was first applied for two dimensional problems (2D), such as flows about a cylinder and an airfoil [18], but later, it was capable of solving three dimensional (3D) problems too. At first, the capacity of early computers and the lack of numerical methods for solving such flows, acted as an obstacle for CFD applications, but in 1970 non-linear effects could be dealt, due to the development of appropriate numerical algorithms and the availability of supercomputers at the same time [19]. The first computers invented (1950) were capable of performing only hundreds of operations per second, in contrast with today's, that reach the value of 10^{12} (tera) operations per second [20]. Also, apart from speed, improvement has been achieved in the capacity of data saved too. Specifically, hard disks with 10^{10} bytes were only available in supercomputers, while nowadays, such numbers are available in personal computers. Historically, the byte was the number of bits used to encode a single character of text in a computer, and for this reason, it is the smallest addressable unit of memory in many computer architectures [21]. Nowadays, CFD is used to solve loads of engineering problems, in the fields of aerodynamics, structural and environmental engineering, fluid flows and heat transfer, combustion processes etc.

1.2.2 Equations solved in CFD

As said above, CFD's purpose is solving the equations that govern the flow of gasses and liquids, with the aim of finding the properties of flow fields. These equations are no other than the conservation equations, such as the continuity, the momentum (Navier-Stokes) and the energy equation. However, due to the fact that in some instances, phenomena like turbulence, combustion and multiphase flows occur, for which either no exact equations are available until now or the numerical solving is not feasible, the induction of models is needed, which will be able to approach real behaviour as accurately as possible [20]. In this section, emphasis will be given on the conservation equations, but before moving on, it would be interesting to make a reference to the methods used to describe the movement of fluids.

1.2.2.1 Description of fluid motion

These methods are the Lagrangian and the Eulerian [22]. In most of the cases, with the former approach discrete systems are described, while continuous ones with the latter. The Lagrangian, is always time-dependent and involves the initial coordinates a, b, c of a particle and its coordinates x, y, z, at time t. Its motion is described by:

$$x = x\,(a, b, c, t),\ y = y\,(a, b, c, t),\ z = z\,(a, b, c, t). \qquad (1.1)$$

The Eulerian, consists of u, v, w, which are the velocity components at point (x, y, z), at time t. The velocity components are the dependent variables with the rest being the independent. The description of motion at time t is:

$$u = u\,(x, y, z, t),\ v = v\,(x, y, z, t),\ w = w\,(x, y, z, t). \qquad (1.2)$$

For the following equations the Euler approach is considered.

1.2.2.2 Conservation of mass

Considering the mass flow through a differential volume dV, as shown in figure 4, and at the same time no transformation of mass into energy and vice versa, the principle of conservation of mass mentions that a mass balance must occur. This means that if no accumulation of mass takes place (e.g., via expansion or compression), then the amount of mass entering the volume should equal the amount exiting it [23]:

[RATE OF MASS ACCUMULATION] = [RATE OF MASS ENTERING THE VOLUME] + [RATE OF MASS EXITING THE VOLUME]. *(1.3)*

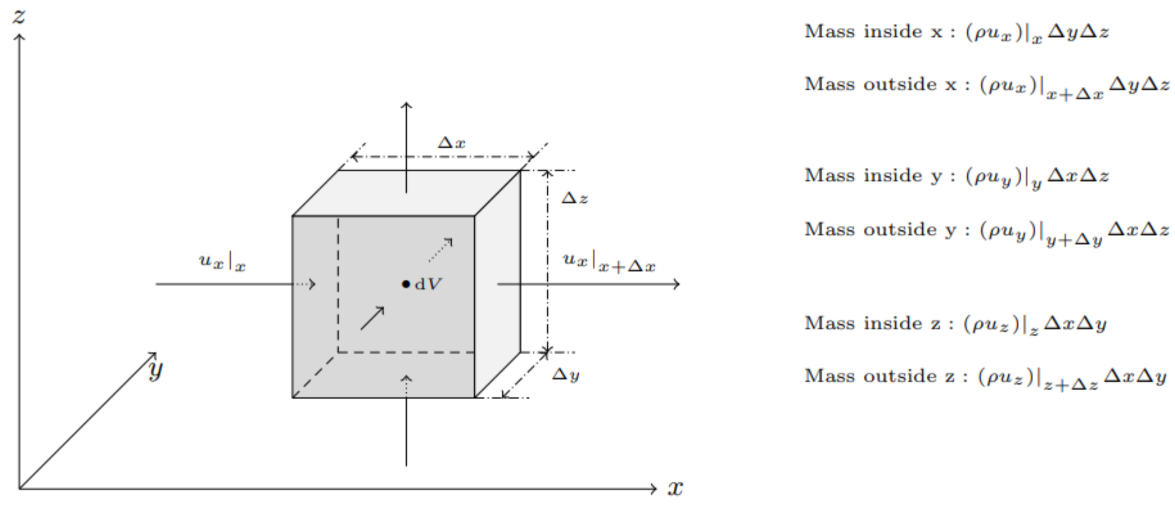

z
y
x
Δx
Δz
Δy
$u_x|_x$
$u_x|_{x+\Delta x}$
$\bullet\,\mathrm{d}V$
Mass inside x : $(\rho u_x)|_x\,\Delta y\Delta z$
Mass outside x : $(\rho u_x)|_{x+\Delta x}\,\Delta y\Delta z$
Mass inside y : $(\rho u_y)|_y\,\Delta x\Delta z$
Mass outside y : $(\rho u_y)|_{y+\Delta y}\,\Delta x\Delta z$
Mass inside z : $(\rho u_z)|_z\,\Delta x\Delta y$
Mass outside z : $(\rho u_z)|_{z+\Delta z}\,\Delta x\Delta y$

Figure 4. *Differential volume dV and mass fluxes [23].*

By applying the above mass balance in the volume, one can obtain the equation, which can be expressed either in a derivative or an integral form. Concerning the derivative form, in which control volume is divided into a large number of small elements, its general expression is:

$$\frac{\partial\rho}{\partial t} = -\nabla\cdot(\rho\boldsymbol{U}) \qquad \textit{(1.4)}$$

where, ∂: partial derivative, ∇: Nabla-Operator, U: velocity, ρ: density, t: time.

If incompressible flow is assumed, the density of the fluid becomes constant and then the expression turns into:

$$\nabla\cdot U = 0. \qquad \textit{(1.5)}$$

At this point, it has to be mentioned that incompressibility means either no expansion or no compression, but still, there can be temperature dependence, so density does not remain constant. However, if very small changes in density are assumed, the above expression can rightly be used. Respectively, by considering the control volume as one element, the general integral form is obtained:

$$\frac{\partial}{\partial t}\int\rho\,dV = -\oint\rho\boldsymbol{U}\cdot\boldsymbol{n}\,dS \qquad (1.6)$$

where, V: volume, d: derivative, $\oint$: convolutional integral, n: normal vector to the surface, S: surface.

In the same way, for the incompressible case, the expression becomes:

$$\oint\rho\boldsymbol{U}\cdot\boldsymbol{n}\,dS = 0. \qquad \textit{(1.7)}$$

1.2.2.3 Conservation of momentum

The momentum equations are no other than the so called «Navier-Stokes» equations and an expression of the Newton's second law for fluids. The equations were derived by Navier, Saint-Venant, Poisson and Stokes between 1827 and 1845 [24] and their main difference with the continuity equations lies in the fact that the momentum quantity is a vector and not a scalar. They are the most well-known equations in the field of fluid mechanics, as by solving them, one can obtain the velocity and pressure profile of a given geometry. However, because of their complexity, it is challenging to obtain analytical solutions, with the most known between them being the one that describes the fluid flow between two parallel plates (Couette flow). Another thing that needs to be mentioned, is that the original Navier-Stokes equations are not appropriate for every flow. Newtonian fluids are those who obey the expression:

$$\tau = \mu \cdot [\boldsymbol{\nabla U} + \boldsymbol{\nabla U}^T] - \frac{2}{3}\mu \boldsymbol{I}(\boldsymbol{\nabla U}) \qquad \textit{(1.8)}$$

where, τ: viscous stress tensor, μ: dynamic viscosity, **I**: identity matrix

Such behaviour is observed in fluids, such as water, air, alcohol etc., whereas some common non-Newtonian fluids are blood, juice, ketchup, etc. Navier-Stokes equations is shown that do not accurately predict flow regimes of non-Newtonian fluids. Also, the equations are only valid for continuum fluids, that is, those whose physical length scale of the system is much larger than the mean free path of the molecules that make up the fluid [24]. Here comes the Knudsen Number:

$$Kn = \frac{\lambda}{L} \qquad \textit{(1.9)}$$

where, λ: mean free path, L: representative length scale.

The equations can be used successfully for Kn<0.01, while for 0.01< Kn <0.1, a special treatment of the boundary conditions is needed. For cases of Kn>0.1, the Navier-Stokes equations cannot be used, and other methods or mathematical models need to be applied.

Moving on to the equations themselves, the momentum balance in a differential volume dV is expressed as follows [23]:

[rate of momentum accumulation] = [rate of momentum entering the volume] – [rate of momentum leaving the volume] + [sum of forces that act on the volume]. *(1.10)*

By applying (1.10) in a differential volume dV in every x, y, z direction, the derivative and integral form is obtained. For the former, the expression in vector form is:

$$\frac{\partial}{\partial t}\rho \boldsymbol{U} = -\boldsymbol{\nabla} \cdot [\rho \boldsymbol{U} \times \boldsymbol{U}] - \boldsymbol{\nabla} \cdot \boldsymbol{\tau} - \boldsymbol{\nabla} p + \rho \boldsymbol{g} \qquad \textit{(1.11)}$$

where, p: static pressure, g: gravitational acceleration, ×: cross product.

In the above expression, the first term represents the momentum convection, the second and third together the viscous force, the fourth the surface force and the last term the mass force. In order to obtain the integral form of the Navier-Stokes equations, the Gauss theorem is used, which states that the volume integral of the divergence over the region inside a surface, is equal to the surface integral of a vector field over a closed surface, i.e., the flux through the surface A [25]:

$$\oiint_S \boldsymbol{A}\, d\boldsymbol{S} = \iiint_V [\boldsymbol{\nabla A}]\, dV. \qquad \textit{(1.12)}$$

So, (1.11) takes the following form:

$$\frac{\partial}{\partial t}\int \rho\boldsymbol{U}\,dV = -\oint [\rho\boldsymbol{UxU}]\cdot\boldsymbol{n}\,dS - \oint \boldsymbol{\tau}\cdot\boldsymbol{n}\,dS - \oint p\boldsymbol{I}\cdot\boldsymbol{n}\,dS + \int \rho\boldsymbol{g}\,dV. \qquad (1.13)$$

1.2.2.4 Conservation of energy

Conservation of energy or in other words the first law of thermodynamics, states that the increase in the system's total energy is a result of the work and heat added to the system:

$$dE_t = dW + dQ \qquad (1.14)$$

where, dW: the work added to the system, dQ: heat added to the system.

Total energy includes both internal (thermal) and kinetic (mechanical) energy, so by applying the energy balance in an arbitrary volume dV, one can get:

[RATE OF INTERNAL AND KINETIC ENERGY ACCUMULATION] = [RATE OF INTERNAL AND KINETIC ENERGY ENTERING THE VOLUME] – [RATE OF INTERNAL AND KINETIC ENERGY LEAVING THE VOLUME] + [NET RATE OF HEAT ADDITION BY CONDUCTION] – [NET RATE OF WORK DONE BY SYSTEM ON SURROUNDINGS] + [NET RATE OF ADDITIONAL HEAT SOURCES] [23]. *(1.15)*

The above expression gives:

$$\rho[\frac{\partial h}{\partial t} + \nabla(hV)] = -\frac{\partial p}{\partial t} + \boldsymbol{\nabla}[k\boldsymbol{\nabla}T] + \Phi \qquad (1.16)$$

where, h: enthalpy, k: thermal conductivity, T: temperature, Φ: dissipation function [26].

The above equation predicts unsteady behaviour and additional heat sources too. Concerning the physical meaning of it, the first term represents the local change with time, the second is the convective term, the third is the pressure work, followed by the heat flux and finally the heat dissipation term. The energy equation is solved when temperature dependent flows are assumed.

1.2.3 The Finite Volume Method

As mentioned before, CFD's purpose is to numerically solve the partial differential equations that govern the motion of fluids. In order to solve them, three main approximation methods exist: the finite difference, the finite volume and the finite element method. In this thesis the finite volume method is going to be addressed. This method's goal is to solve the conservation equations, which is clear that they have significant commonalities. Specifically, by introducing the general variable φ, the conservative form of all fluid flow equations can be derived, taking the following form:

$$\frac{\partial(\rho\varphi)}{\partial t} + \boldsymbol{\nabla}(\rho\varphi\boldsymbol{U}) = \boldsymbol{\nabla}(\Gamma\boldsymbol{\nabla}\varphi) + S_\varphi \qquad (1.17)$$

where, Γ: diffusion coefficient, S_{φ}: source term [27].

Equation *(1.17)* is the so-called transport equation for property ϕ and in words can be expressed as:

[RATE OF INCREASE OF φ OF FLUID ELEMENT] + [NET RATE OF FLOW OF φ OUT OF FLUID ELEMENT] = [RATE OF INCREASE OF φ DUE TO DIFFUSION] + [RATE OF INCREASE OF φ DUE TO SOURCES]. (1.18)

The above equation, once again contains different transport processes, among them, the rate of change of the term *φ*, followed by the convection term, the diffusion term and finally the source term. In order to see the commonalities mentioned, the non-shared terms of the conservation equations are hidden in the source term.

The method considers the transport equation as the starting point, and according to the problem in relation, by setting the values of *φ, Γ* and S_{φ}, it is simplified in specific forms. The main goal of the discretization process is to calculate the property ϕ at the control volume faces and the fluxes across the domain's boundaries. Starting off, equation *(1.17)* is integrated over a three-dimensional control volume CV, and by applying the Gauss Theorem, the volume integral turns into a surface integral:

$$\frac{\partial}{\partial t}\left(\int_{C\gamma}\rho\varphi\,dV\right)+\int_{A}\boldsymbol{n}(\rho\varphi\boldsymbol{U})\,dA=\int_{A}\boldsymbol{n}(\Gamma\nabla\varphi)\,dA+\int_{CV}S_{\varphi}\,dV. \quad (1.19)$$

The above equation represents the flux balance in a control volume and it is this specific equation that is discretized, in order to be converted into an algebraic system and be solved. It should be mentioned that, *(1.19)* is applied to a finite number of finite control volumes, which if summed up, make up the whole computational domain, as shown in figure 5.

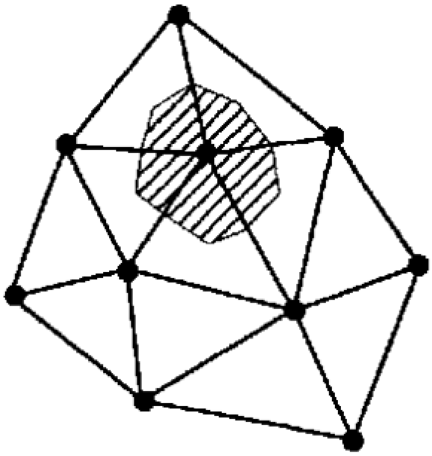


Figure 5. *A computational domain, in which there is a finite control volume (grey area) [28].*

Each finite volume consists of nodal points, and as indicated in figure 6, the faces of the volumes are positioned mid-way between adjacent nodes. Node *P* is a general nodal point, which at the same time constitutes the discretization point, and *W, E* are its neighbor nodes. Moving on, *w, e* are the finite control volume's boundaries, while the finite volume's length is $\Delta x = \delta x_{we}$.

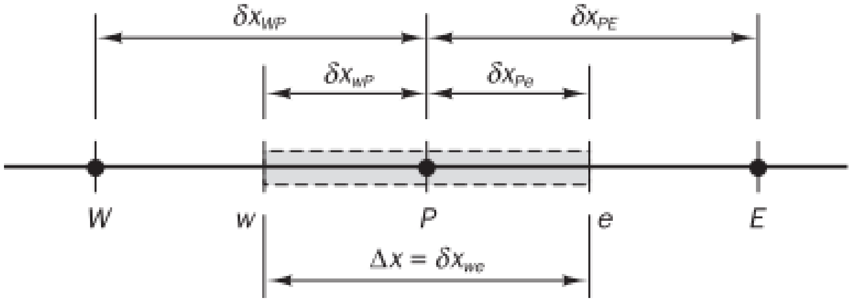


Figure 6. *Finite control volume and nodal points [27].*

Considering for simplicity, a one-dimensional flow field *u*, with the absence of sources, then the steady state convection and diffusion equation of a property *φ* is:

$$\frac{d}{dx}(\rho u \varphi) = \frac{d}{dx}\left(\Gamma \frac{d\varphi}{dx}\right). \tag{1.20}$$

The above equation should also abide by the continuity equation:

$$\frac{d(\rho u)}{dx} = 0. \tag{1.21}$$

Integration of *(1.20)* over the control volume of figure 6 yields:

$$(\rho u A \varphi)_e - (\rho u A \varphi)_w = \left(\Gamma A \frac{d\varphi}{dx}\right)_e - \left(\Gamma A \frac{d\varphi}{dx}\right)_w. \tag{1.22}$$

Integration of *(1.21)* yields:

$$(\rho u A)_e - (\rho u A)_w = 0. \tag{1.23}$$

The final form of the discretized algebraic equations is obtained by introducing two new variables *F, D,* that represent the convective mass flux per unit area and the diffusion conductance at cell faces:

$$F = \rho u\,,\ D = \frac{\Gamma}{\delta x}.$$

Replacement of F, D yields the final equation:

$$F_e \varphi_e - F_w \varphi_w = D_e(\varphi_E - \varphi_P) - D_w(\varphi_P - \varphi_W). \tag{1.24}$$

For the continuity equation the discretized form is:

$$F_e - F_w = 0. \tag{1.25}$$

Assuming that the velocity field is known (known F_e and F_w), in order to solve *(1.24*) property ϕ at faces *e* and *w* need to be calculated. Schemes for this purpose follow next, but because of their big number, reference will be made to the two most important between them, which are the central differencing and the upwind differencing schemes.

Starting off, the central differencing approximation can be used to represent the diffusion terms of the right-hand side of the equation *(1.24),* while for the convective terms of the left-hand side of the same equation, a linear interpolation is conducted. So, the cell face values of property ϕ are written as:

$\varphi_e = (\varphi_P + \varphi_E) / 2.$ *(1.26)*

$\varphi_w = (\varphi_W + \varphi_P) / 2.$ *(1.27)*

Finally, the central differencing expressions can be obtained:

$a_P\varphi_P = a_W\varphi_W + a_E\varphi_E$ *(1.28)*

where, a_W: $D_W + F_W / 2$, a_E: $D_e - F_e/2$, a_P: $a_W + a_E + (F_e - F_W)$.

In order to solve the problem and obtain the transported property ϕ, one has to solve the above discretized equation *(1.28)* for all grid nodes.

Moving on to the upwind differencing approximation, its major advantage over the central differencing approximation is its ability to identify flow direction, when calculating the value at a cell face. For the specific scheme, when the flow is in the positive direction, i.e., $u_W > 0$, $u_e > 0$, the scheme sets:

$\varphi_w = \varphi_W$ and $\varphi_e = \varphi_P$. *(1.29)*

So, the equation obtained is:

$[(D_W + F_W) + D_e + (F_e - F_W)]\ \varphi_P = (D_W + F_W)\ \varphi_W + D_e\varphi_E.$ *(1.30)*

For the flow in the negative direction respectively, the scheme sets:

$\varphi_w = \varphi_P$ and $\varphi_e = \varphi_E$. *(1.31)*

This time the discretized equation takes the form:

$[D_W + (D_e - F_e) + (F_e - F_W)]\ \varphi_P = D_W\varphi_W + (D_e - F_e)\ \varphi_E.$ *(1.32)*

Again, the upwind differencing expressions can be obtained:

$a_P\varphi_P = a_W\varphi_W + a_E\varphi_E$ *(1.33)*

where, $a_P = a_W + a_E + (F_e - F_W)$, $a_W = D_W + max\ (F_W\ ,\ 0)$ and $a_E = D_e + max\ (0\ ,\ - F_e)$.

The above coefficients cover both flow directions and again equation *(1.33)* needs to be solved for all grid nodes in order to get the property ϕ.

1.3 Turbulence modeling

1.3.1 The nature of turbulence

All fluid flows in nature, therefore jet flows too, divide into two main categories: laminar and turbulent flows. These two types of flow were first reported by the British mathematician Osborne Reynolds (1883) [29], who carried out the following experiment: With the use of a simple device, as shown in figure 7, Reynolds poured dye inside a pipe, in which water flowed due to gravity. After the experiment, he made the below observations:

- For relatively low volume flow rates, the dye followed an almost straight, without perturbations, trajectory.
- For high volume flow rates, the dye dissipated in a chaotic way, on the whole flow’s cross-section.

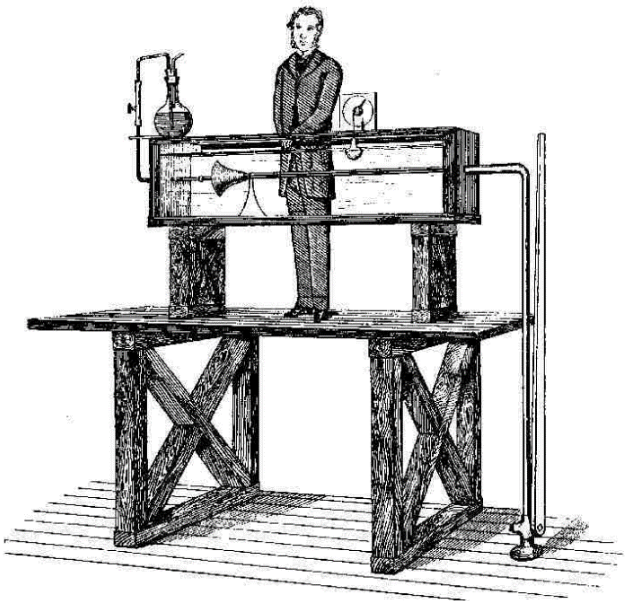


Figure 7. *Osborne Reynolds’ experiment device [30].*

These two states, are no other than the laminar and the turbulent flow, respectively, and their difference is illustrated in figure 8. In the rest of this section, emphasis will be given on turbulent flows, due to their frequency in nature and the fact that this thesis deals with turbulence in impinging jets. Speaking of turbulence, it must be mentioned that there are numerous opportunities to observe turbulent flows in everyday life, beginning from the simplest, like a pipe flow and cases like water getting out of a tap, smoke from a chimney, wind around an airfoil or a motorbike etc.

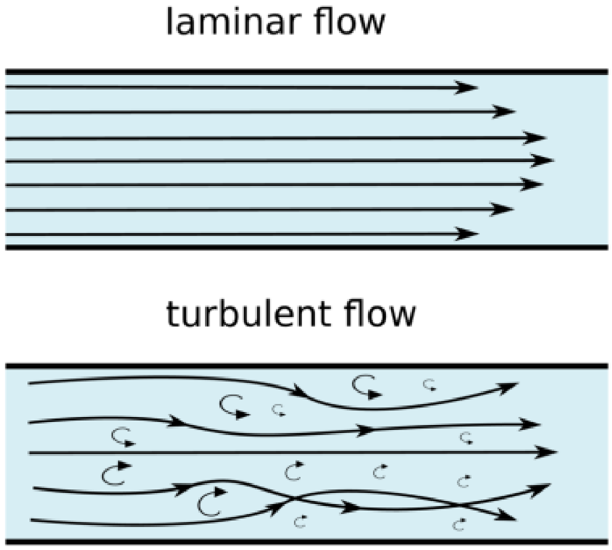


Figure 8. Difference between a laminar and a turbulent flow field [31].

What remains to be mentioned is how the discrimination between the two flow types is done. This is accomplished via the dimensionless Reynolds Number (Re), whose range determines the flow type. Specifically, Reynolds number is defined as:

$$R_e = \frac{\rho UL}{\mu} = \frac{UL}{v} \tag{1.34}$$

where, L: characteristic length scale, μ: dynamic viscosity, ν: kinematic viscosity [29].

At this point, it is useful to say that for pipe flows, therefore at jet flows too, the scale L is replaced by the pipe's diameter. A flow is considered to be laminar if Re<3500 and turbulent if Re>4000. The intermediate range is called developing region and the flow characteristics in it is a mixture of the characteristics of the two main types. Closing the chapter of turbulence, it would be useful to refer to the key attributes of turbulent flows. Such flows usually occur when the velocity or the characteristic length is increased, so the convective forces in the flow overcome the viscous forces [32]. After all, this is the physical meaning of the Reynolds number. Apart from that, vorticity is present and therefore high diffusivity is present too, which results in rapid mixing and high energy transfer [33].

1.3.2 The Reynolds-Averaged Navier-Stokes approach

Although there are plenty of approaches in literature to account for turbulent flows, in this thesis only the Reynolds-Averaged Navier-Stokes (RANS) approach will be examined. Among other methods, it is assumed to be the most cost-effective [32], because the degrees of freedom required to be solved are less than those of the other methods'. For the same reason, it is suitable for most of the most complex geometries. In general, it is a numerical method, which is used to model turbulent flows and in which flow variables are decomposed into time-averaged and fluctuating components [35]. This technique is used, because in turbulent flows, the velocity and pressure terms that take place in the momentum and pressure equations vary with time, due to turbulent fluctuations. The technique is called «Reynolds decomposition», and is illustrated in figure 9. What it accomplishes is that it time-averages the Navier-Stokes equations and this way, the complexity of the system's partial differential equations is reduced. Mathematically the solution is written as:

$$u(x,t) = \bar{u}(x) + u'(x,t) \tag{1.35}$$

where, $\bar{u}(x)$: mean flow velocity, $u'(x,t)$: time varying fluctuations in velocity, x: position vector, t: time.

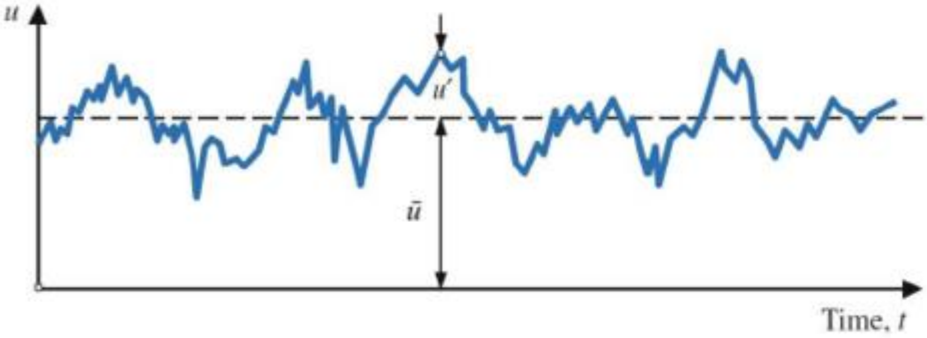


__Figure 9.__ Reynolds decomposition for the velocity u [36].

At this point, it is useful to rewrite the continuity and momentum equations, respectively, in notation form:

$$\frac{\partial \rho}{\partial t} + \frac{\partial}{\partial x_i}(\rho u_i) = 0. \tag{1.36}$$

$$\frac{\partial}{\partial t}(\rho u_i) + \frac{\partial}{\partial x_j}(\rho u_i u_j) = -\frac{\partial p}{\partial x_i} + \frac{\partial}{\partial x_j}\left(\rho\mu \frac{\partial u_i}{\partial x_j}\right). \tag{1.37}$$

After the decomposition, and by applying *(1.35)* to *(1.36)* and *(1.37)*, the following non-linear equations are obtained:

$$\frac{\partial(\rho \overline{u_j})}{\partial x_j} = 0. \tag{1.38}$$

$$\frac{\partial(\rho \overline{u_i})}{\partial t} + \frac{\partial}{\partial x_j}(\rho \overline{u_i}\overline{u_j}) = -\frac{\partial \bar{p}}{\partial x_i} + \mu\left(\frac{\partial \bar{u}_i}{\partial x_j}\frac{\partial \bar{u}_i}{\partial x_j}\right) - \frac{\partial}{\partial x_j}(\rho \overline{u_i' u_j'}) \tag{1.39}$$

where, $\bar{p}$: mean pressure.

Special attention should be given to the term $\rho\overline{u_i' u_j'}$, which is a non-linear term that accounts for the anisotropy of turbulence and is called the Reynolds stress (some authors refer to $u_i' u_j'$ as Reynolds stress). It physically stands for the mean forces (per unit area) imposed on the mean flow by turbulent fluctuations [37]. The term is expressed as:

$$\overline{u_i'u_j'} = \begin{pmatrix} \overline{u'^2} & u'v' & u'w' \\ v'u' & \overline{v'^2} & v'w' \\ w'u' & v'w' & \overline{w'^2} \end{pmatrix}. \tag{1.40}$$

In this matrix, the diagonal terms make up the normal stresses, while the rest make up the shear stresses and apparently there are six independent variables. The system cannot be closed directly by equations for Reynolds stresses, because of the unknown stress terms, so it is necessary to replace those unknown Reynolds stress terms with mean flow quantities [38]. A renowned solution to the closure problem described above is the Eddy Viscosity Method, which is based on the Boussinesq hypothesis (1887) [39], that relates the Reynolds stresses with the mean velocity gradient:

$$\tau_{ij} = -\rho\overline{u_i'u_j'} = \nu_t\left(\frac{\partial\overline{u_i}}{\partial x_j} + \frac{\partial\overline{u_j}}{\partial x_i}\right) - \frac{2}{3}\left(\rho k + \nu_t\frac{\partial\overline{u_i}}{\partial x_i}\right)\delta_{ij} \tag{1.41}$$

where, ν_t: turbulent eddy viscosity, k: turbulent kinetic energy, δ_{ij}: Kronecker delta term.

The variables ν_t, k are defined as followed:

$$\nu_t = \rho c_\mu V_t l_t \tag{1.42}$$

where, c_μ: a turbulence model constant, V_t: eddy velocity scale, l_t: eddy length scale.

$$k = \frac{1}{2}\overline{u_i'u_i'}. \tag{1.43}$$

Finally, by substitution of *(1.41)* into *(1.39)*, the equation becomes:

$$\frac{\partial(\overline{u_i})}{\partial t} + \frac{\partial}{\partial x_j}\left(\overline{u_i u_j}\right) = -\frac{1}{\rho}\frac{\partial}{\partial x_i}\left(\bar{p} + \frac{2k}{3}\right) + \frac{\partial}{\partial x_j}\left((\nu + \nu_t)\left(\frac{\partial(\overline{u_i})}{\partial x_j}\right)\right). \tag{1.44}$$

The above equation is the one that all of the turbulence models are based.

1.3.3 Turbulence models

Moving on, it’s now time to go back to section 1.2.2, where a reference to turbulence models was made. It was then mentioned, that various phenomena, one of which is turbulence, demand extreme computational resources to be fully resolved. As a result, turbulence models have been introduced. Generally, the «turbulence problem» is considered notoriously challenging and the solution approach is based on the power of computers [34]. Difficulties arise for numerous reasons. First of all, the velocity field U (x, t) is three-dimensional, time-dependent and random and its motion is dependent on the geometry boundaries, because of the largest motions whose range reach the boundaries. In addition, difficulties come from the non-linear convective term of the Navier-Stokes equations and the pressure-gradient term. These difficulties are dealt with methodologies that include partial differential equations and sometimes additional algebraic equations, which in combination with the appropriate initial and boundary conditions, are solved numerically. Mathematically, the goal of turbulence models is to solve for v_t, and substitute to *(1.44).*

1.3.3.1 The k-ω SST turbulence model

The k-ω Shear Stress Transport (SST) turbulence model was first proposed by Menter (1994) as a two- equation eddy viscosity model [40]. It is a combination of the k-ε and the k-ω turbulence models and exploits the advantages of both. The k-ε is a two-equation model that solves for two variables: k, the turbulence kinetic energy and ε, the rate of dissipation of turbulence kinetic energy. In the k-ω model, ε is replaced by ω, the specific rate of dissipation of kinetic energy. Specifically, in the free-stream it is transformed into the k-ε model, while near the walls, takes the form of the k-ω model. This way, it avoids the unreliability of the near-wall damping functions that the former uses, and the sensitivity of the latter to the free-stream values of ω.

Wall functions are used as a bridge, which connects the inner-region of the wall and the fully developed turbulent region of the flow. They are about empirical equations which are used to satisfy the physics of the flow near the walls, and in which, the first cell center must be located inside the log law-region. Their use significantly reduces the computational demands. The turbulent flow over a flat plat can be divided into four regions, as shown in figure 10. Before moving on, it would be useful to refer to two dimensionless quantities that are important for the definition of the below regions: u^+ and y^+. Speaking of the former, it is about a dimensionless distance from the wall that is defined as:

$$y^+ = \frac{yu_\tau}{v} \tag{1.45}$$

where, y: distance to the wall, u_τ: friction or shear velocity, v: kinematic velocity [41].

Friction velocity is defined as:

$$u_\tau = \sqrt{\frac{\tau_w}{\rho}} \tag{1.46}$$

where, τ_w: wall-shear stress.

Speaking of the latter, it is defined as:

$$u^+ = \frac{u}{u_\tau} \tag{1.47}$$

where, u: velocity u parallel to the wall.

At first, a thin layer above the wall is formed in which the velocity is linear with the distance from the wall, creating the so-called viscous sublayer [42]. This region locates below 5 wall units ($y^+ < 5$) and the expression obeyed is:

$$u^+ = y^+. \tag{1.48}$$

Next follows the buffer-layer ($5 < y^+ < 30$), in which the turbulence stresses begin to dominate over the viscous ones and which connects with the log-law region which is almost fully turbulent and the average velocity is a log-function of the distance to the wall. There ($y^+ > 30$), the appropriate expression becomes:

$$u^+ = \frac{1}{\kappa} ln\, y^+ + C^+ \tag{1.49}$$

where, κ: von Karman constant, C^+: a constant.

Finally, even further away follows the free-stream region. By using wall-functions, one ignores the flow in the buffer layer and analytically computes the flow in the viscous sublayer, so computational costs become much less.

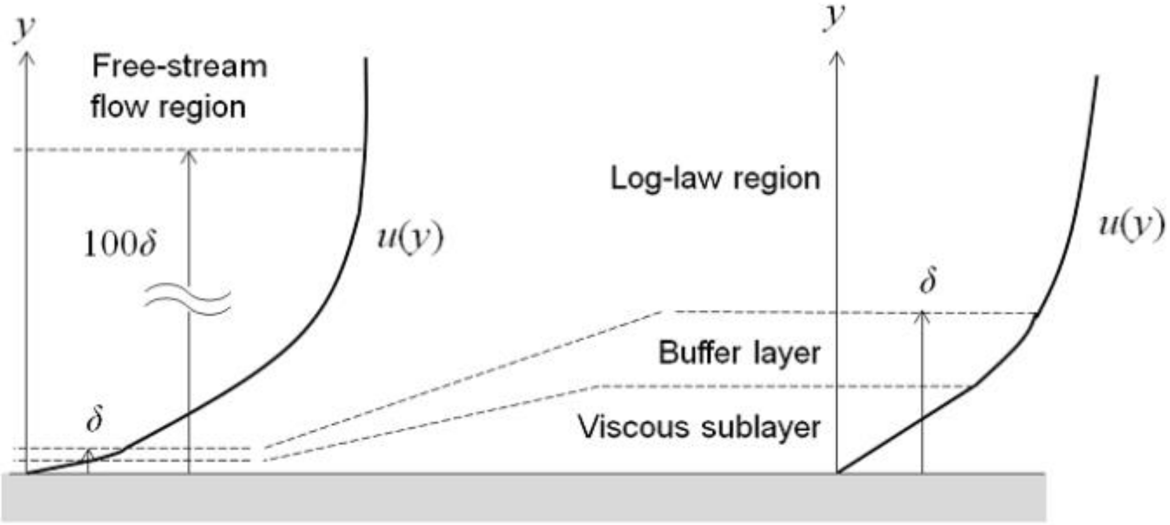


Figure 10. Regions of the flow over a flat plate [42].

Moving on to the two equations of the model, these are the following:

$$\frac{\partial k}{\partial t} + U_j \frac{\partial k}{\partial x_j} = P_k - \beta^* k\omega + \frac{\partial}{\partial x_j}\left[(\nu + \sigma_k \nu_T)\frac{\partial k}{\partial x_j}\right]. \tag{1.50}$$

$$\frac{\partial \omega}{\partial t} + U_j \cdot \frac{\partial \omega}{\partial x_j} = aS^2 - \beta\omega^2 + \frac{\partial}{\partial x_j}\left[(\nu + \sigma_\omega \nu_T)\frac{\partial \omega}{\partial x_j}\right] + 2(1 - F_1)\sigma_{\omega 2} \cdot \frac{1}{\omega}\frac{\partial k}{\partial x_i}\frac{\partial \omega}{\partial x_i}. \tag{1.51}$$

More information about the different variables and constants of the equations can be found in literature [43]. At this point, it is worth mentioning that the alteration between the two original models is done via the blending function F_1, which, for the value of $F_1 = 0$, turns the model into the k-ε model, for the value of $F_1 = 1$, turns it into the k-ω model and for intermediate values, creates a smooth transition between the two models in the intermediate cells of the domain. By solving the above equations, one can find k and ω, and afterwards find v_t and substitute it in *(1.44)*, via the below formulas [44]:

$$k = \frac{3}{2}(UI)^2 \tag{1.52}$$

where, U: mean flow velocity, I: turbulence intensity.

$$\varepsilon = C_\mu \cdot \frac{k^{3/2}}{l} \tag{1.53}$$

where, C_μ: a constant, l: turbulent length scale.

$$\omega = \frac{\varepsilon}{C_\mu \cdot k}. \tag{1.54}$$

$$v_T = \frac{k}{\omega}. \tag{1.55}$$

Turbulent length scale is defined as:

$$l = 0.038 D_h \tag{1.56}$$

where, D_h: hydraulic diameter.

1.4 Diploma thesis organization

The rest of this thesis splits into four parts, taking over chapters 2-5. Specifically, in chapter 2, a 2D mesh dependency study was conducted with the aim of validating experimental findings existing in literature. A test of the impact of 2D simulations on 3D results was done, and the effect of the y^+ value of the impingement surface was investigated. All of the results illustrate the velocity of the fluid. Next, in chapter 3, axisymmetric turbulent jets were numerically studied with the aim of improving our understanding of their behaviour in the near-field impingement. A 3D steady-state simulation and the same 3D transient simulation was run, and the results were verified and validated with existing findings. This time, the results involved velocities, wall shear stresses and Reynolds stresses. In chapter 4, a parametric analysis of the cases of chapter 3 was done, where the effect of various velocities and kinematic viscosities (through the variation of Reynolds number) on the development of the jet flow field near the impingement surface was studied. The results concerned the same properties with chapter's 3. In the last 5th chapter, some general conclusions followed, and afterwards, potential future work involving multi-phase impinging jets was addressed.

Unit 2. **TWO-DIMENSIONAL SIMULATIONS**

2.1 Simulations in OpenFOAM

OpenFOAM (Open-Source Field Operation and Manipulation) [45] is a freely available open-source package developed in *C++* programming language [46], with the aim of developing customized numerical solvers, as well as pre/post-processing utilities for solving mainly fluid dynamics and continuum mechanics problems [47]. Specifically, solutions involving chemical reactions, heat transfer, turbulence, acoustics, solid mechanics and electromagnetics can be found [48]. *OpenFOAM* consists of a series of *C++* libraries that are used with the aim of creating executables, called applications. On their turn, applications divide into solvers and utilities, with the latter used for performing tasks involving data manipulation. An overview of *OpenFOAM* structure can be seen in figure 11.

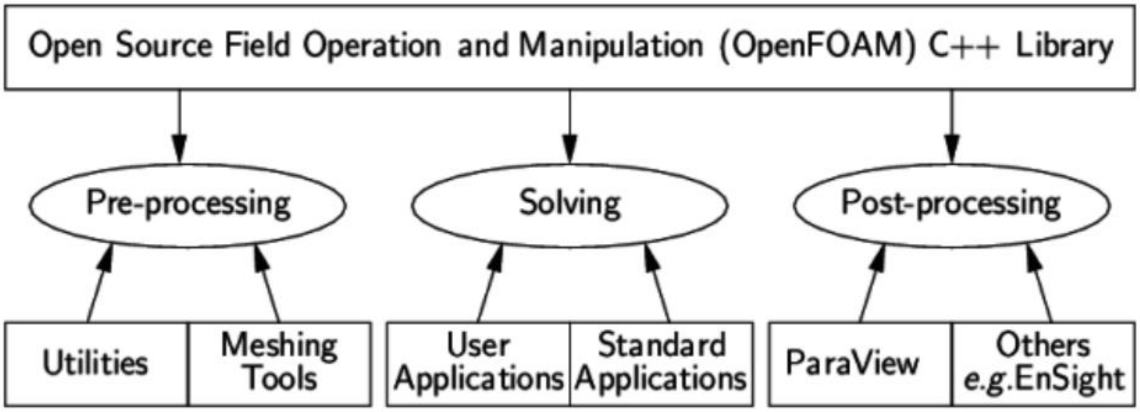


Figure 11. *OpenFOAM structure [49].*

Instead of using proprietary CFD software, *OpenFOAM* is used in this study as it has the main advantage of being a free-to-use and free-to-modify modern high-end CFD code. As a result, the development and programming of new applications, solvers and utilities is becoming much easier. Besides the great advantages presented above, the software has the flaw of requiring a good *C++* knowledge background. Among the different versions, in this thesis *OpenFOAM 5.x* is going to be implemented.

In order to run a simulation at *OpenFOAM*, a directory structure consisting of different files exists. In addition, it must be clarified that *OpenFOAM* contains multiple solvers, which may require more or less files than the others. Nevertheless, the incompressible solvers that will be used in this thesis, require three main files: the «0», the «constant» and the «system» files. With their turn, these files consist of subfolders, shown in table 1. This table includes the files that are necessary when solving a turbulent flow only for pressure and velocity fields.

Table 1. *Files and subfolders of a solver that solves only pressure and velocity [50].*

0	Constant	System
U	transportProperties	controlDict
P	turbulentProperties	fvSchemes
		fvSolutions

Speaking of the time directories (0), they contain files with data of various fields. These data are both initial and boundary conditions of the different variables of the problem solved. It must be specified that if one wants to initialize a simulation with the latest results of another (e.g. initialize a transient simulation with the latest time-step of the steady state), then the «0» file is replaced by that time-step. Concerning the «constant» directory, it contains the physical properties of the fluid, such as its kinematic viscosity and, after importing the mesh, its full description is also included through the subdirectory «polyMesh». Apart from that, it can also contain turbulence properties, if turbulence is considered. The «system» directory contains parameters that have to do with the solution procedure itself. The first of the three main files, «controlDict», is where run control parameters are set, such as start time, end time, time-step and other output settings. The second one, «fvSchemes», includes all the discretization schemes used in the solution, while «fvSolutions», includes the solvers, the tolerances and other algorithm controls [51].

The first simulations carried out by *OpenFOAM*, had the aim of validating the results that Khaled J. Hammad and Ivana M. Milanovic found through their experimental investigation [12]. Although an experiment is a real-life simulation, and as a result it is three-dimensional (3D), the 2D simulations aimed at testing whether or not this difference played an important role at the validation of the results. As a result, a meshing dependency test was conducted in order to examine the impact of the mesh resolution on the results.

2.2 Experimental setup

The experimental setup is illustrated in figure 12. The system consists of three main components: the test section, the pump and the pipe mount. The cylindrical test section, which is housed in a square tank, is made of smooth finish clear cast acrylic and has a diameter of $D_{cylinder}$= 101.6 mm, a height of $H_{cylinder}$= 279.4 mm and wall thickness of 3.175 mm. A submerged vertical tube made of 304 stainless steel is used to produce the jet and guide it downwards. Its geometric characteristics are: an inner diameter of D= 6.35 mm, a length of L= 711.2 mm and wall thickness of 0.381 mm, while the length-to-diameter ratio of the tube, L/D= 122, assured fully developed flow conditions at the outlet. After the impingement on the bottom wall, the flow leaves the test section through four symmetrically distributed circular holes of diameter D_{hole}= 6.33 mm, located 20.8 mm below the top plate, as shown in figure 13. Experiments were conducted for eight pipe-to-plate distances, for the values from 1-8.

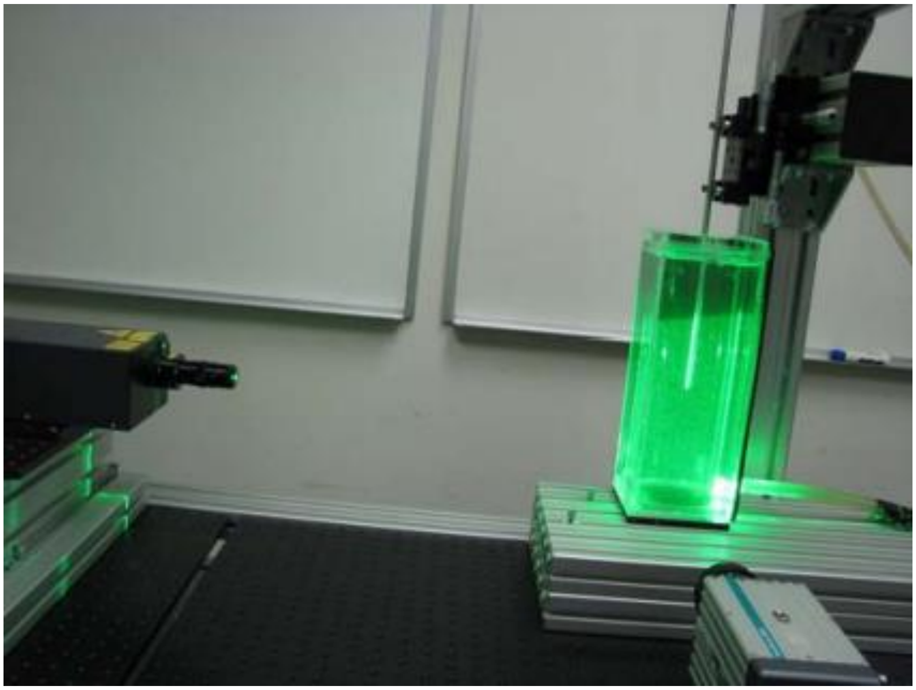


Figure 12. Experimental setup [12].

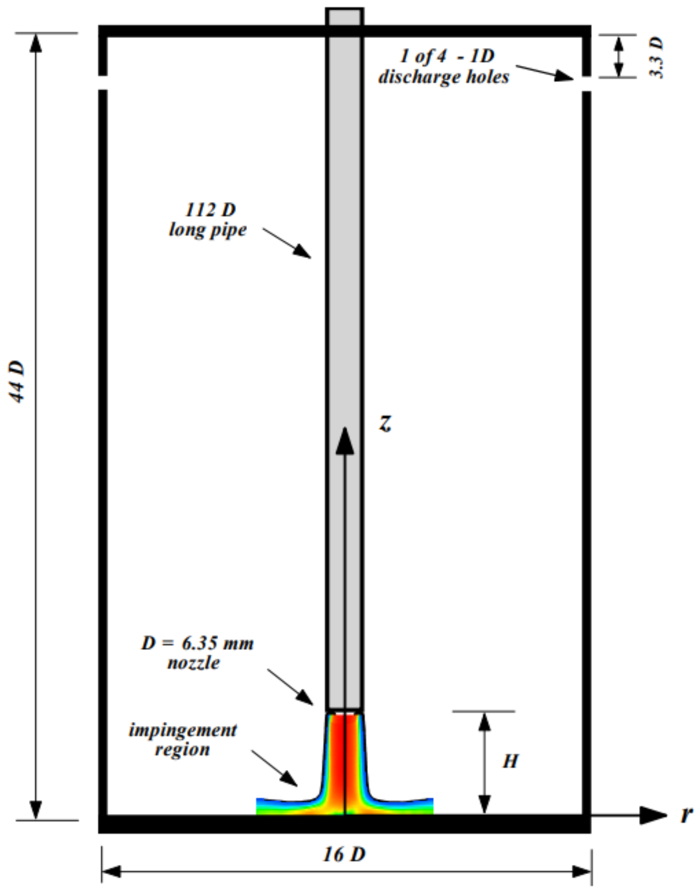


__Figure 13.__ Test section schematic [12].

The flow properties of the experiment are listed in table 2, below. As working fluid, water was used.

__Table 2__. Flow properties [12].

Volumetric flow rate	6.37×10^{-5} m^3/s
Pipe bulk velocity	2.01 m/s
Temperature	30 °C
Kinematic viscosity	8.03×10^{-7} m^2/s
Reynolds number	15,895

For the experiment itself, a two-dimensional Particle Image Velocimetry (PIV) system was used in order to measure the flow field at each r-z plane, which is an optical measurement technique that measures the velocity field of an entire region [52]. The system consists of a dual-cavity laser, a camera, a synchronization hardware for controlling the laser and camera, and software running on a Windows-based platform for the data management and post-processing. More details on the characteristics of the components and the way they were used for the specific experiment can be found on the paper. The study focused on the investigation of the flow structure near the impingement region, as a result, the findings concern the region defined by $|r/D|\leq3$ and $0<z<D$. Concerning velocity, the mean velocity is considered for this case, which is expressed as:

$$U = \frac{1}{N} \cdot \sum_{i=1}^{N} u_i \qquad (2.1)$$

where, u_i is the measured instantaneous velocity and N is the number of PIV data sets.

2.3 Geometry of the simulations

Moving on to the simulations themselves, a differentiation in the geometry was done, which however, should not have an impact on the results. In more detail, in order to reduce more the complexity of the geometry, because of the fact that it was not created in a computer-aided design (CAD) application, the four outlet holes were not considered, but instead, the outlet was the free upper surface, as shown in figure 14. The visualization of the geometry was done via P*araview* [53], an open-source data-analysis and visualization platform. Also, it must be clarified that the inlet pipe sticks out some distance above, but could not be illustrated here. The geometry is made up of six boundary faces: the «inlet», the «outlet», the «pipe», the «left», the «right» and the «bottom» faces, also illustrated below.

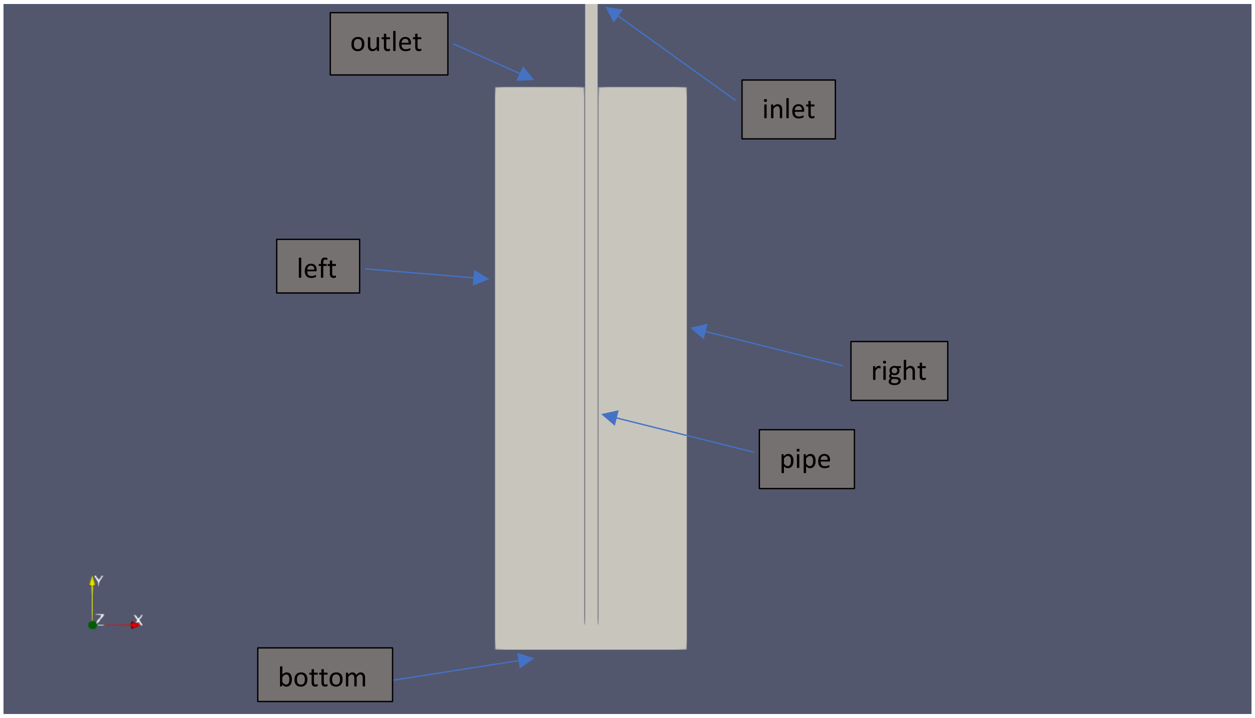
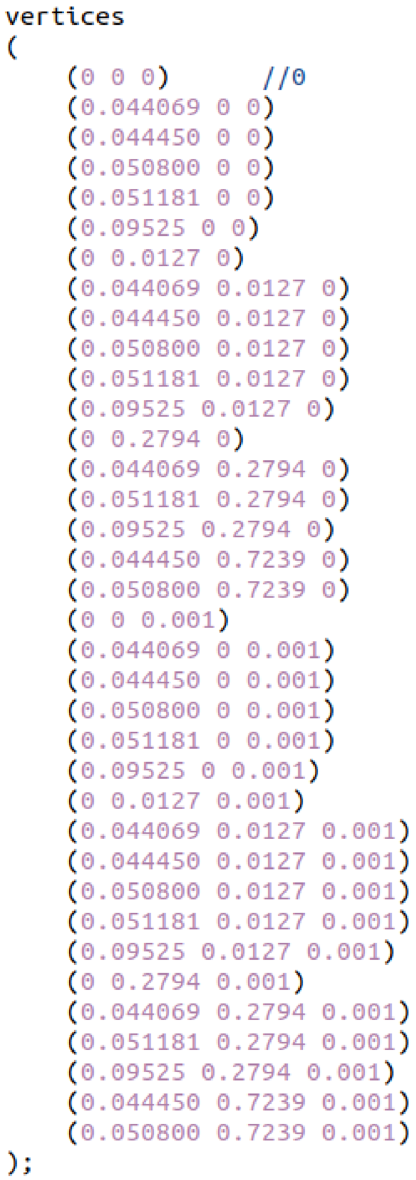


Figure 14. *Geometry with boundary faces.*

The geometry was created via the dictionary file «blockMeshDict», a geometry and mesh generation utility supplied with *OpenFOAM*. This directory is located in the *constant/polyMesh* directory of a case. At first, the geometry is created via vertices, as shown in figure 15 and then is divided into hexahedral blocks, in which different mesh grading can be applied. In our case, the geometry was divided into 8 blocks and apparently, the geometric lengths were identical to the experiment’s. The nozzle-to-plate distance that was considered was, H/D=2.



```
vertices
(
    (0 0 0)      //0
    (0.044069 0 0)
    (0.044450 0 0)
    (0.050800 0 0)
    (0.051181 0 0)
    (0.09525 0 0)
    (0 0.0127 0)
    (0.044069 0.0127 0)
    (0.044450 0.0127 0)
    (0.050800 0.0127 0)
    (0.051181 0.0127 0)
    (0.09525 0.0127 0)
    (0 0.2794 0)
    (0.044069 0.2794 0)
    (0.051181 0.2794 0)
    (0.09525 0.2794 0)
    (0.044450 0.7239 0)
    (0.050800 0.7239 0)
    (0 0 0.001)
    (0.044069 0 0.001)
    (0.044450 0 0.001)
    (0.050800 0 0.001)
    (0.051181 0 0.001)
    (0.09525 0 0.001)
    (0 0.0127 0.001)
    (0.044069 0.0127 0.001)
    (0.044450 0.0127 0.001)
    (0.050800 0.0127 0.001)
    (0.051181 0.0127 0.001)
    (0.09525 0.0127 0.001)
    (0 0.2794 0.001)
    (0.044069 0.2794 0.001)
    (0.051181 0.2794 0.001)
    (0.09525 0.2794 0.001)
    (0.044450 0.7239 0.001)
    (0.050800 0.7239 0.001)
);
```

Figure 15. *A part of the BlockMeshDict, in which the geometry is defined via vertices.*

2.4 Mesh dependency test

In the first simulations carried out, four different meshes were examined. In figures 16, 17, 18, 19 follow the four meshes close to the region of interest, which is no other than the region close to the impingement surface. To create the mesh, the geometry was first divided into blocks, in which different number of cells and different grading were added. Each mesh was generated with the «blockMesh» command via the Ubuntu's terminal [54], consisting of hexahedra cells. The quality of the mesh was then checked via the «checkMesh» command at Ubuntu's terminal, which checks the max aspect ratio and the max skewness of the cells, as well as the mesh's non-orthogonality [55]. The mesh was thicker closer to the impingement surface and coarsened radially. As z coordinates, the value of 0.001m was introduced, which is a very small value, appropriate for turning the simulation into 2D. The k-ω SST was considered to account for turbulence, as the Re=15,895 signifies turbulent regime (Re>4000) [33]. The model also allowed for the use of wall-functions that reduced the computational cost. As a result, two of the meshes used the high-Reynolds form (wall functions) of the k-ω SST model, while the other two, its low-Reynolds form (no wall-functions). Consequently, the impact of the y^+ value of the bottom face was also evaluated. In table 3 follow some details with respect to the different meshes.

Table 3. Different meshing details.

Mesh number	Number of cells	k-ω SST form	y^+ value in bottom face
1	9,480	High-Reynolds	37
2	35,250	High-Reynolds	7
3	240,960	Low-Reynolds	5.5
4	440,080	Low-Reynolds	1.5

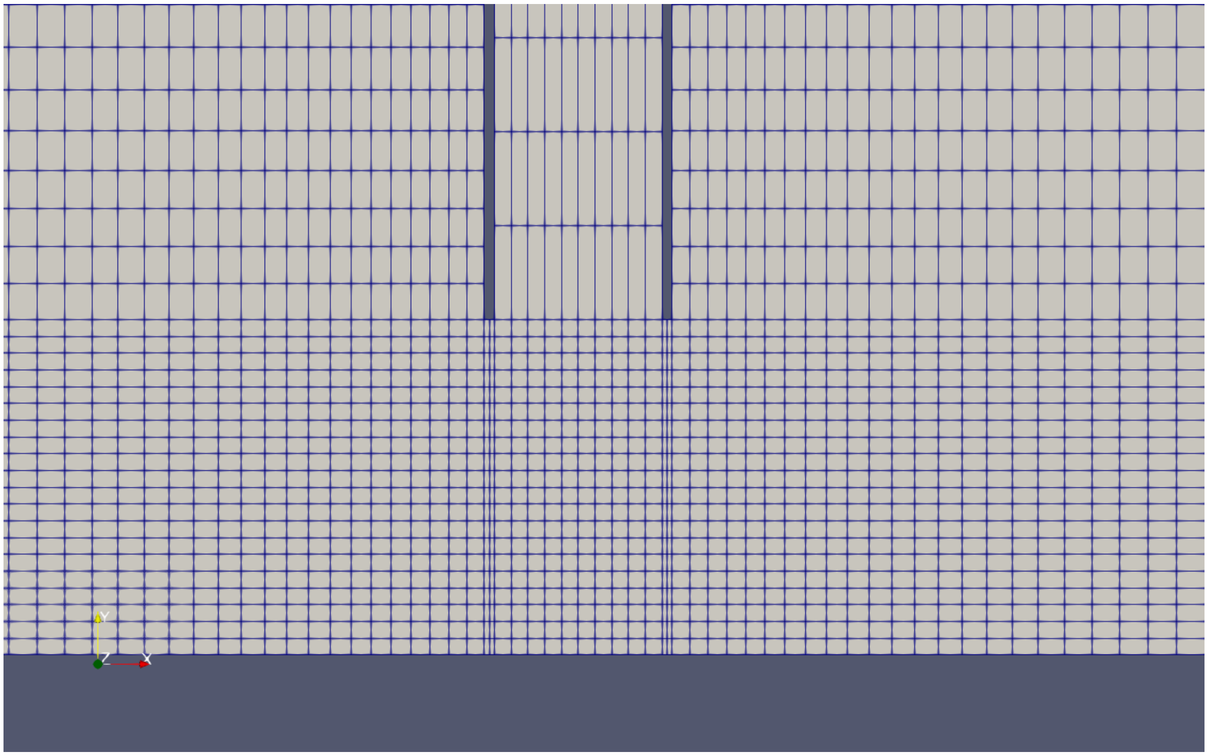


Figure 16. *The mesh with the 9,480 cells.*

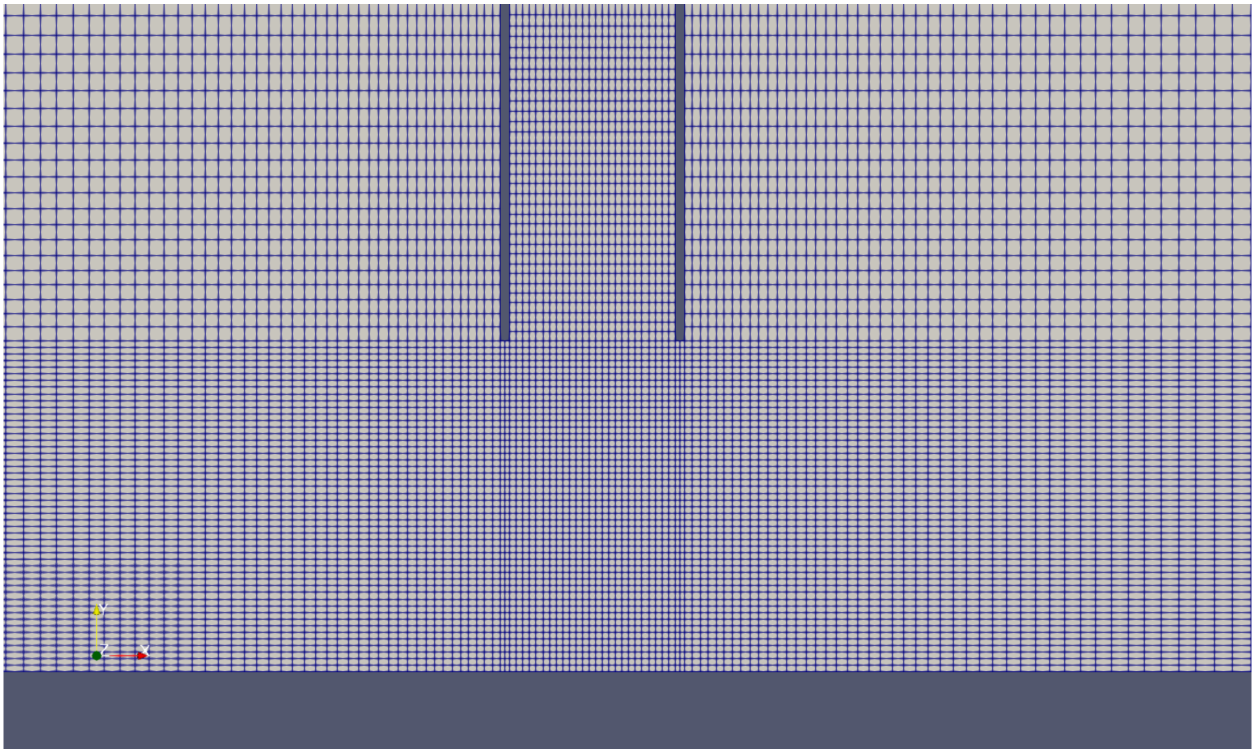


Figure 17. *The mesh with the 35,250 cells.*

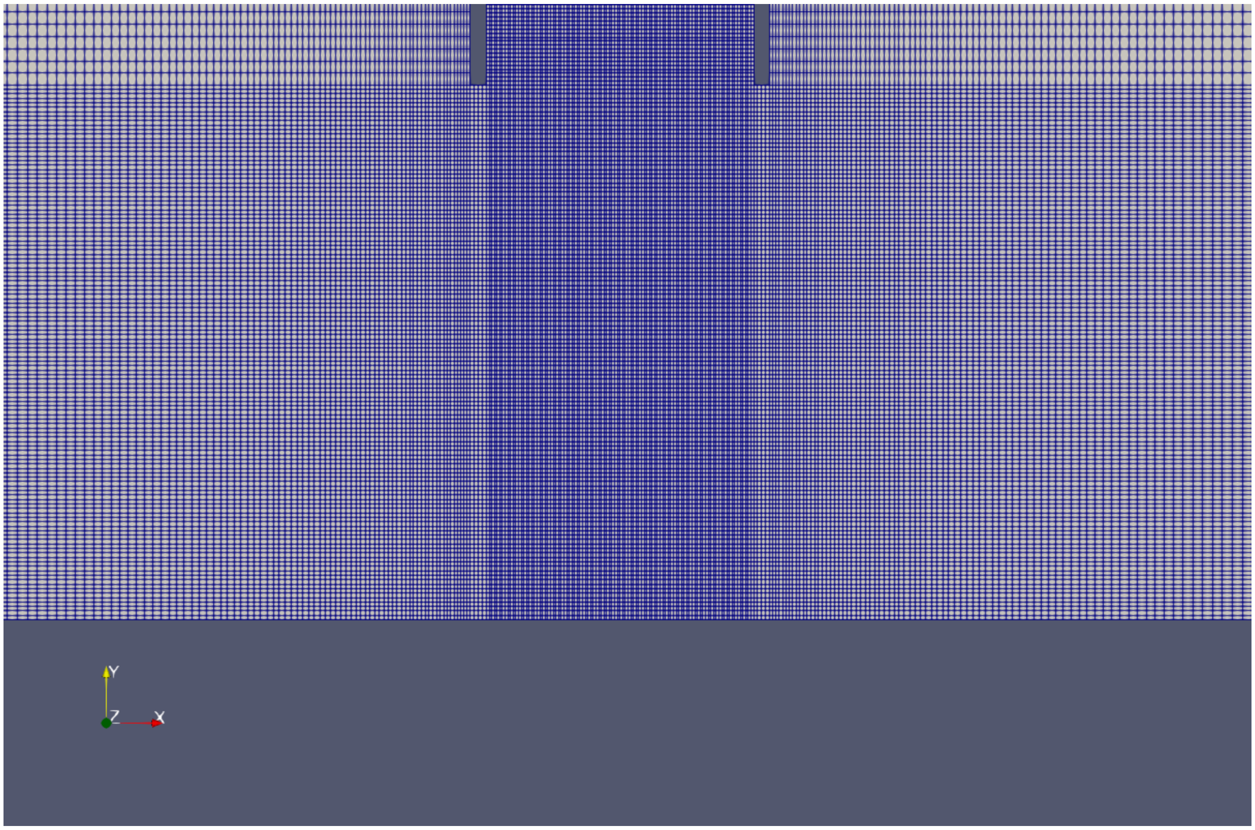


Figure 18. *The mesh with the 240,960 cells.*

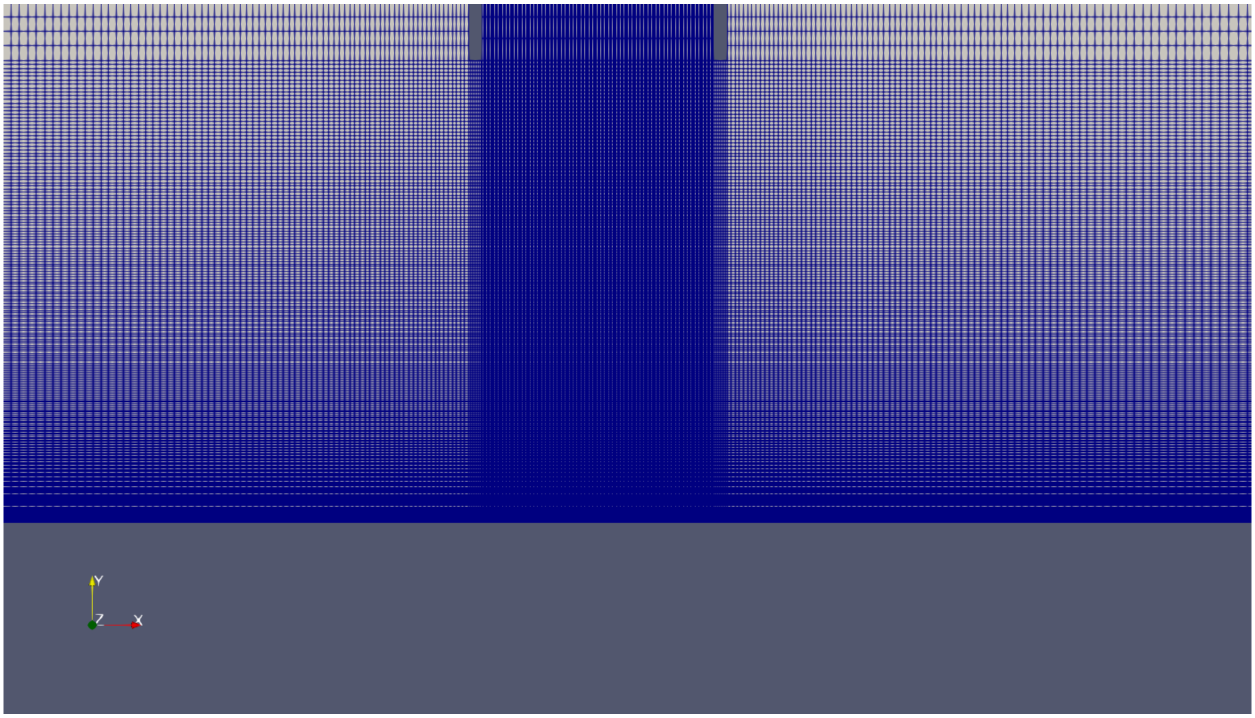


Figure 19. *The mesh with the 440,080 cells.*

Special interest lies between meshes 2 and 3, because for almost the same y^+ values, different wall treatment was used. In addition, a y^+ value of $5<y^+<30$ is known from literature to have a first cell height inside the buffer layer, which for many authors is said to give inaccurate results. However, F. Menter *et al.* has shown that for the k-ω SST model, the whole $0<y^+<300$ range can accurately be applied [56], with the use of the appropriate wall-functions for the high-Reynolds form of the model. So, a comparison with the other meshes' results was made.

Due to the steady-state, turbulent nature of the problem, as well as its incompressibility, the solver «simpleFoam» was used. This solver, uses the SIMPLE (Semi-Implicit Method for Pressure Linked Equations) algorithm, which is an iterative algorithm that numerically solves the Navier-Stokes equations, following a segregated solution strategy, meaning that the equations for each variable involved (pressure p, velocity U and turbulent variables) are solved sequentially and the solution of the preceding equations is inserted in the subsequent equations. For further details on the algorithm, the reader can search in the recommended literature [57] and references therein.

What's more, for each geometry face both initial and boundary conditions were imposed. Initial conditions were the same for all of the four meshes, but boundary conditions differed. That was imperative, due to the different forms of the turbulence model. Concerning the turbulence intensity, the value of 5% was chosen arbitrarily, as the real value cannot be estimated exactly experimentally. However, this is considered a realistic value, existing in flows with relatively low velocities and simple geometries [58]. For k, omega and nut, *(1.52)-(1.56)* were used, where for the constant C_μ, the value of 0.09 should be acquired according to literature. In table 4, the initial conditions are gathered, while in table 5 follow the boundary conditions for the first mesh. Boundary conditions for U, p and ω were the same for all four meshes, while different conditions were applied in meshes 3 and 4 (low-Reynolds form) for k and ε, in «pipe» and «bottom» faces. There, «fixedValue 1e-11» was used for k and «fixedValue 2.05209e-05» for nut. Also, not much attention was paid to the y+ values of the other wall faces, as they were located far enough from the region of interest, which was no other than the jet impingement region, and at the same time, what happened in that area did not affect the values in the impingement location. In incompressible solvers, OpenFOAM's pressure value is in fact the pressure divided by the fluid density (p/ρ), so the units become m^2/s^2.

Table 4. *Initial conditions.*

U (m/s)	p (m^2/s^2)	k (m^2/s^2)	omega (1/s)	nut (m^2/s)
(0 0 0)	0	0.0151	512.435	2.05209e-05

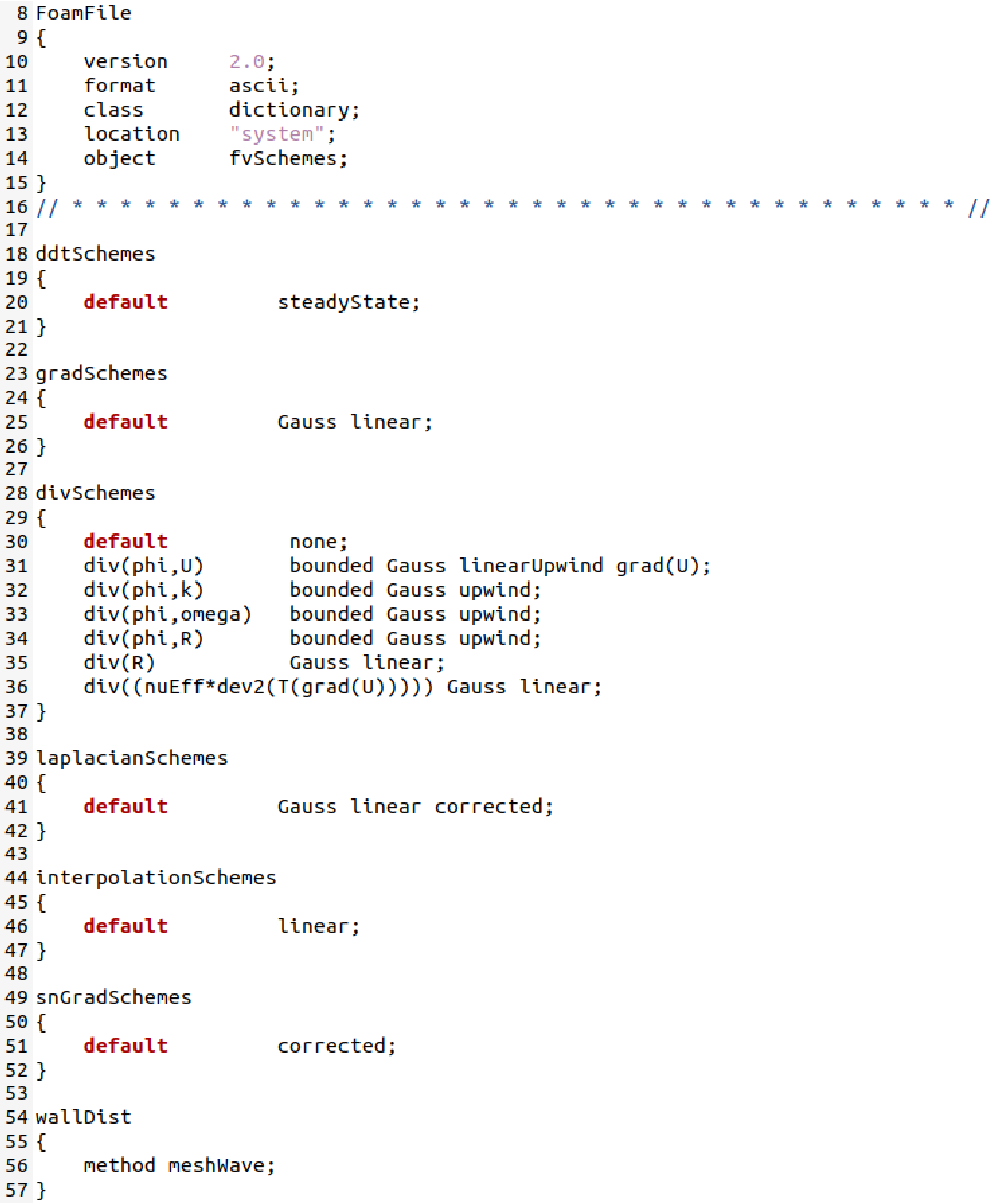
Table 5. *Boundary conditions for the first mesh.*

	inlet	outlet	pipe	left	right	bottom
U (m^2/s)	fixedValue (0 -2.01 0)	zeroGradient	noSlip	noSlip	noSlip	noSlip
p (m^2/s^2)	zeroGradient	fixedValue 0	zeroGradient	zeroGradient	zeroGradient	zeroGradient
k (m^2/s^2)	fixedValue 0.0151	zeroGradient	kqRWallFunction 0.0151	kqRWallFunction 0.0151	kqRWallFunction 0.0151	kqRWallFunction 0.0151
omega (1/s)	fixedValue 512.435	zeroGradient	omegaWallFunction 512.435	omegaWallFunction 512.435	omegaWallFunction 512.435	omegaWallFunction 512.435
nut (m^2/s)	fixedValue 2.05209e-05	fixedValue 2.05209e-05	nutkWallFunction 2.05209e-05	nutkWallFunction 2.05209e-05	nutkWallFunction 2.05209e-05	nutkWallFunction 2.05209e-05

The «kqRWallFunction» boundary condition inherits the traits of the «zeroGradient» boundary condition [59]. As far as the value of 1e-11 for the low-Reynolds bottom boundary condition, instead of attributing the value of 0 for the turbulence kinetic energy on the walls, a very small value like this is suitable for simulation purposes. «nutkWallFunction» sets the turbulent viscosity in the first node point based on the logarithmic law. In the case of «omega», particular interest lies in the fact that unlike the other turbulence variables, boundary conditions are the same between the meshes. This is due to the wall function «omegaWallFunction», which depending on the local y+ values, can switch the formulas between those used in the viscous and the logarithmic regions [60]. The «noSlip» boundary condition is the same as using the «fixedValue (0 0 0)» condition, which ensures there is no fluid movement on the walls' surfaces.

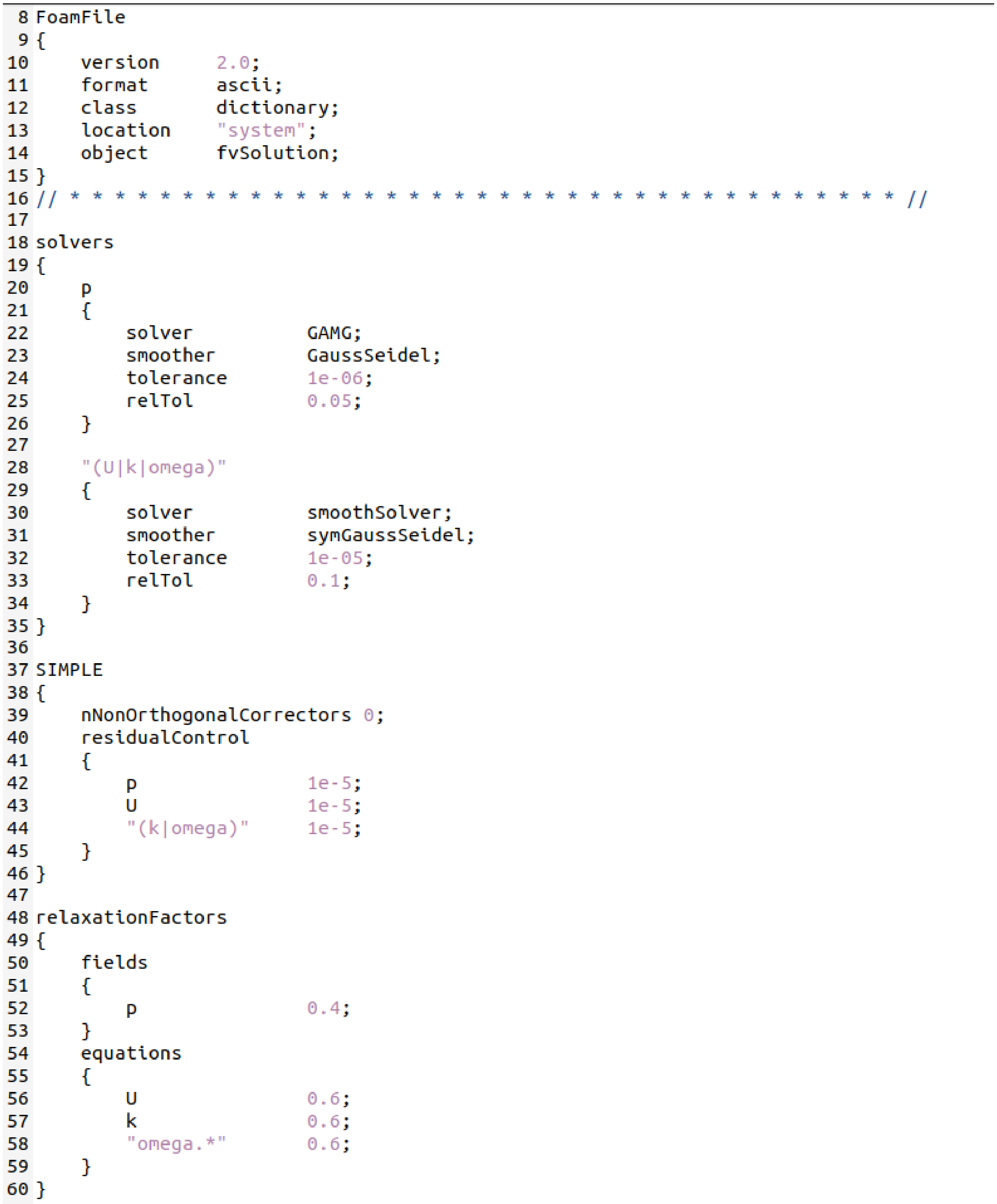
Subsequently, the discretization schemes of the simulations had to be determined, which are gathered in figure 20. The schemes were the same for all the meshes. Again, more information about the schemes can be found on *OpenFOAM's* user guide [61]. The schemes used were a mix of 1st and 2nd order schemes, with the former guaranteeing stability and the latter, accuracy. The word «bounded» was added, in order to ensure boundedness of the solution and to avoid convergence issues [62], while the «meshWave» method was used as a calculation of distance from the wall faces.

In figure 21, the «fvSolutions» directory is shown. This directory differs only for the first mesh, in which for the relaxation factors the values of 0.4 were chosen for pressure and 0.6 for the other variables, while the value of 0.5 was chosen for all the variables in the rest of the meshes. Those factors' purpose is to limit the amount a variable changes during a solution step. After numerous trial and errors, these values seemed to give the best convergence and stability results in the least computational cost. The residuals of the algorithm were set to the value of 1e-5. That value represents the value the initial residuals of the field equations must fall, in order for the algorithm to be considered to have converged and subsequently terminate. Initial residuals on their turn, represent a measure of the stability of the solution with respect to the previous iteration, while final residuals that involve in the tolerances definition, have to do with the change of the solution of the equations, after the solver has terminated. The absolute tolerance was chosen 1e-6 for pressure and 1e-5 for velocity, which is the value the final residual must fall in order for the solver to stop. Relative tolerance is defined as the relation between the initial and final residual of a variable, in order for the solver to stop. The values chosen were 0.05 for pressure and 0.1 for velocity. More details about the solvers and the other settings can be found in guides [63].



```
FoamFile
{
    version     2.0;
    format      ascii;
    class       dictionary;
    location    "system";
    object      fvSchemes;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //

ddtSchemes
{
    default         steadyState;
}

gradSchemes
{
    default         Gauss linear;
}

divSchemes
{
    default          none;
    div(phi,U)       bounded Gauss linearUpwind grad(U);
    div(phi,k)       bounded Gauss upwind;
    div(phi,omega)   bounded Gauss upwind;
    div(phi,R)       bounded Gauss upwind;
    div(R)           Gauss linear;
    div((nuEff*dev2(T(grad(U))))) Gauss linear;
}

laplacianSchemes
{
    default         Gauss linear corrected;
}

interpolationSchemes
{
    default         linear;
}

snGradSchemes
{
    default         corrected;
}

wallDist
{
    method meshWave;
}
```

Figure 20. *Discretization schemes for the 2D simulations in the OpenFOAM fvSchemes dictionary.*



```
FoamFile
{
    version     2.0;
    format      ascii;
    class       dictionary;
    location    "system";
    object      fvSolution;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //

solvers
{
    p
    {
        solver          GAMG;
        smoother        GaussSeidel;
        tolerance       1e-06;
        relTol          0.05;
    }

    "(U|k|omega)"
    {
        solver          smoothSolver;
        smoother        symGaussSeidel;
        tolerance       1e-05;
        relTol          0.1;
    }
}

SIMPLE
{
    nNonOrthogonalCorrectors 0;
    residualControl
    {
        p               1e-5;
        U               1e-5;
        "(k|omega)"     1e-5;
    }
}

relaxationFactors
{
    fields
    {
        p               0.4;
    }
    equations
    {
        U               0.6;
        k               0.6;
        "omega.*"       0.6;
    }
}
```

Figure 21. *Solution and algorithm control for the 1st mesh in the OpenFOAM fvSolutions dictionary.*

The simulations for the four different meshes were assumed to have converged after 7,000, 15,000, 10,748 and 27,000 iterations for the first, second, third and fourth mesh, respectively. Although the residuals did not reach the value of 1e-5, which was the initial condition for the solver to stop, their cathodic and constant course in combination with their adequately low values was assumed to be enough to consider convergence. The highest residuals in all the meshes were for the pressure, taking values from 1e-3 to 1e-4. All other variables' residuals were between 1e-4 and 1e-6.

2.5 Results of the simulations

In this section the results of the different simulations are presented. After exporting the data from *Paraview*, the plots were created in *LibreOffice*, a free office suite [64].In all the diagrams that follow, velocity plots are illustrated for the case of H/D=2. Results include either axial or radial velocities divided by the centerline axial velocity U_C=2.48 m/s, at different locations of the domain. This way the results become dimensionless. In order to depict the experimental results from the paper's diagrams into the thesis' diagrams, the software *Engauge Digitizer* [65] was implemented. This software is a tool that accepts images containing graphs and recovers their data. The continuous lines correspond to the simulations' results, while the dotted ones, to the experimental.

Starting off, in figure 22, the flow field is indicatively given in order to have a general perspective of the global flow field inside the geometry. In figure 23 the radial profiles of axial velocity, obtained 0.3D away from the outlet is illustrated for the different meshes. This ratio was compared with the experimental findings. Concerning the coordinate system, this is the same that is used in figure 14.

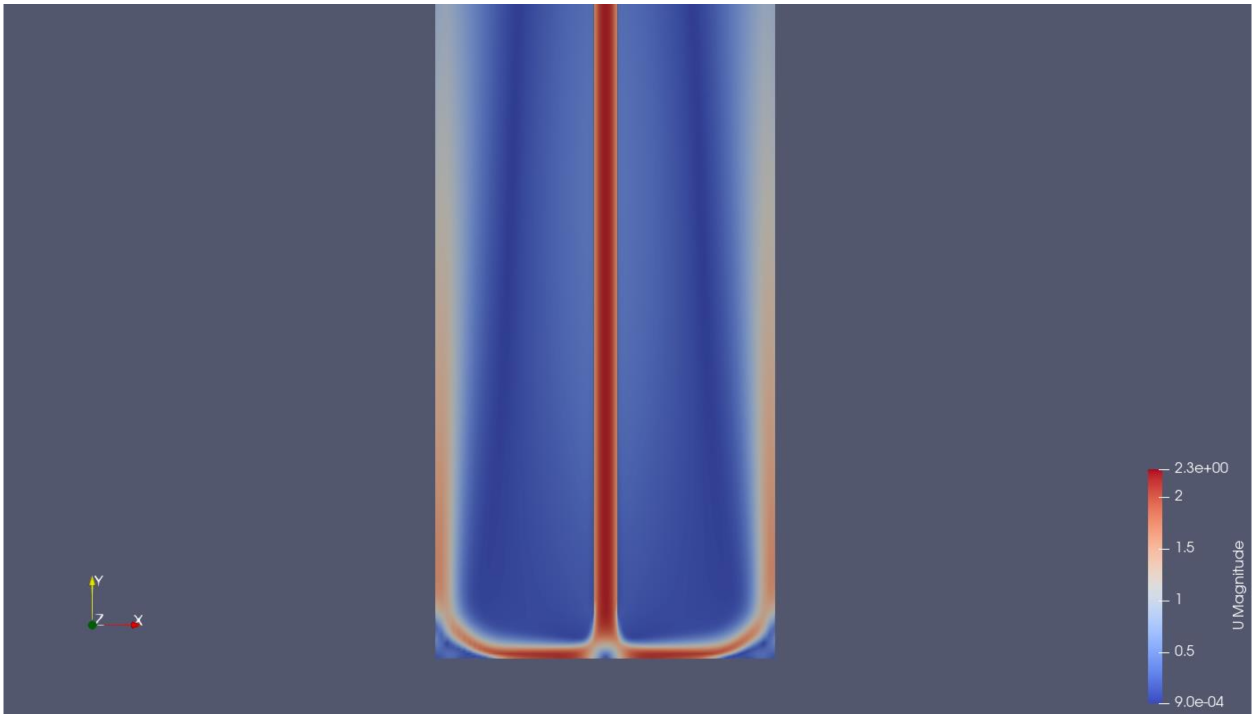


Figure 22. *Flow field of the simulation done with the first mesh, in Paraview's environment.*

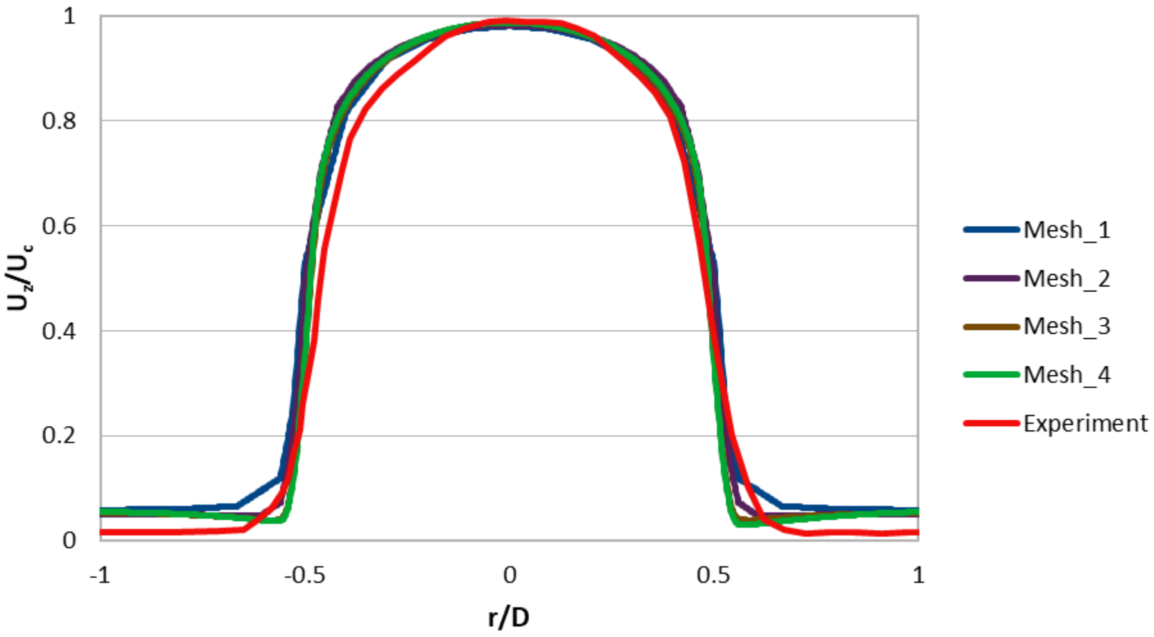


Figure 23. *Mean axial velocity as a function of radial distance for the four meshes and the experiment, 0.3D away from the outlet.*

Respectively, in figure 24 follows the mean axial velocity along the stagnation line.

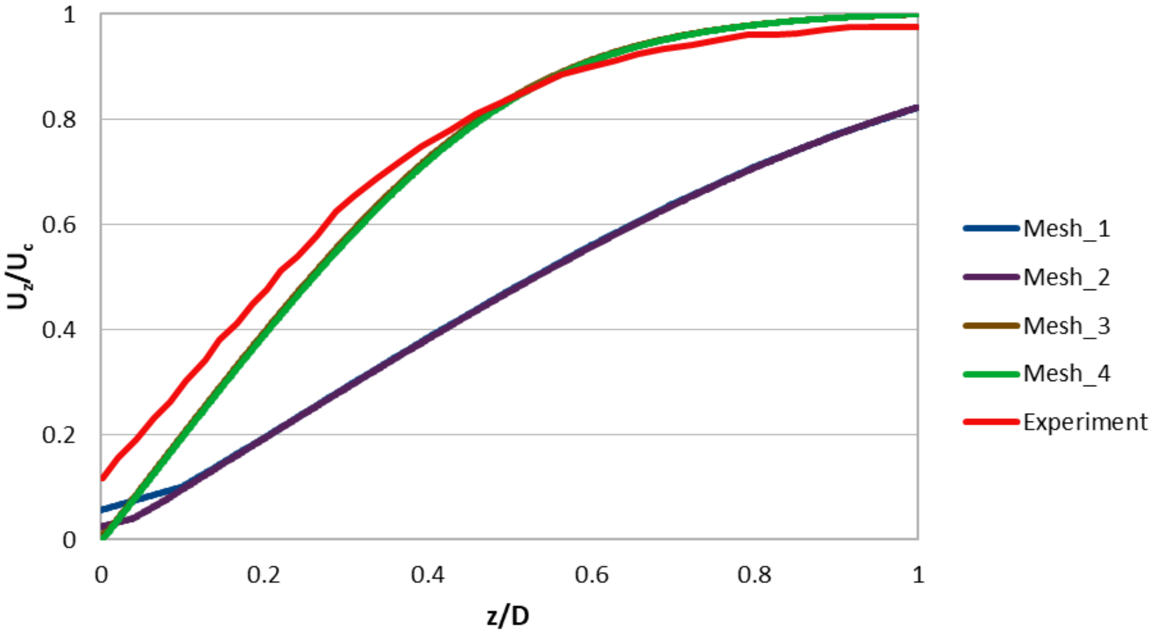


Figure 24. *Mean axial velocity along stagnation line for the four meshes and the experiment.*

Moving on to the figures 25, 26, 27 and 28, there the mean radial velocity profiles at different axial distances from the impingement plate are shown.

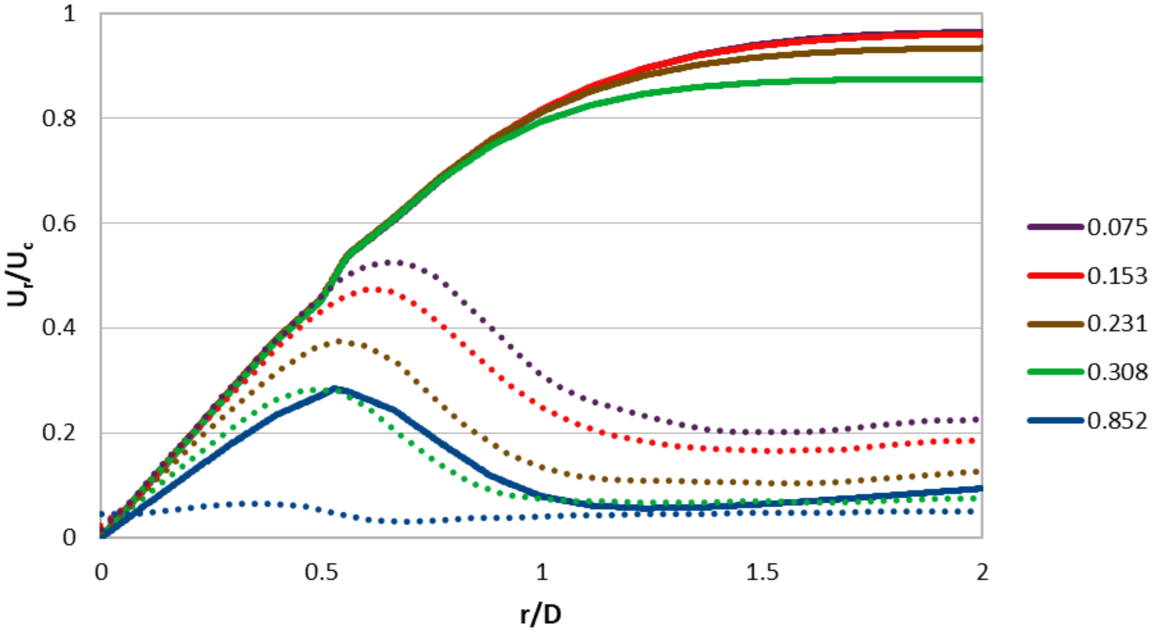


Figure 25. *Mean radial velocity profiles at different axial distances from the impingement plate for the first mesh and the experiment.*

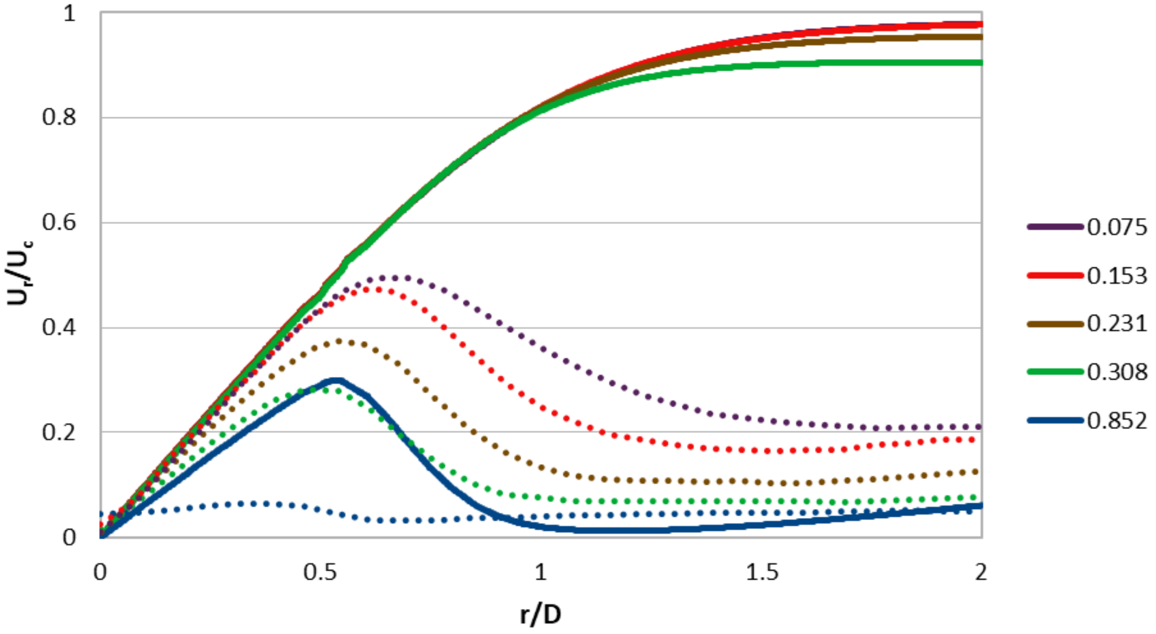


Figure 26. *Mean radial velocity profiles at different axial distances from the impingement plate for the second mesh and the experiment.*

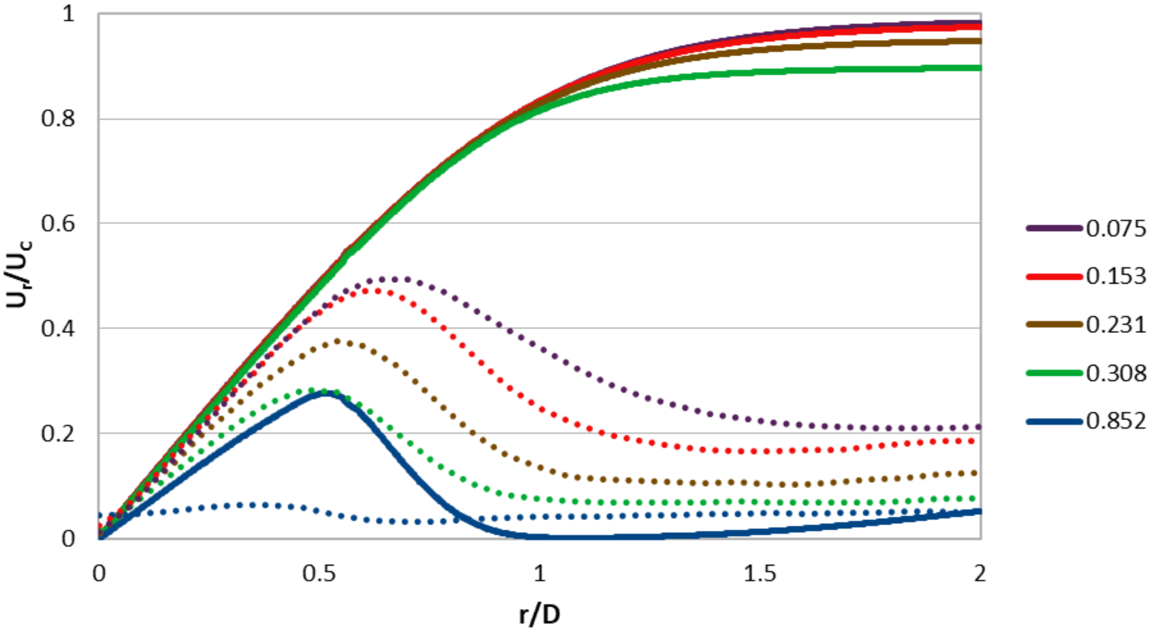


Figure 27. *Mean radial velocity profiles at different axial distances from the impingement plate for the third mesh and the experiment.*

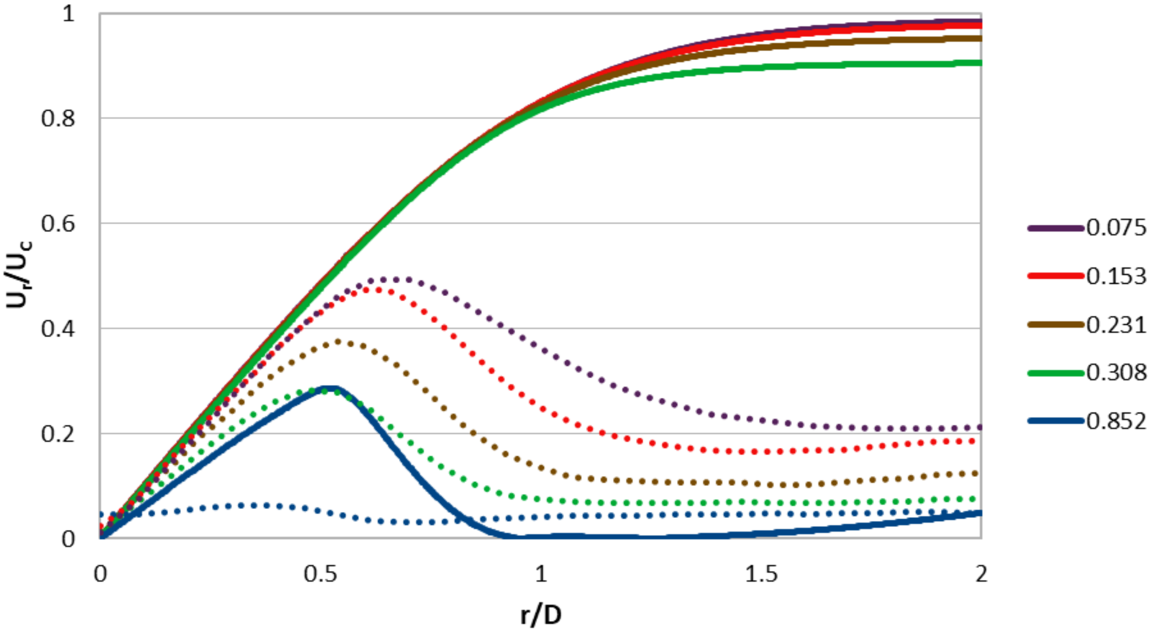


Figure 28. *Mean radial velocity profiles at different axial distances from the impingement plate for the fourth mesh and the experiment.*

The figures 29, 30, 31, 32 on their turn, illustrate the mean axial velocity profiles at different axial distances from the impingement plate, for the four meshes and the experiment.

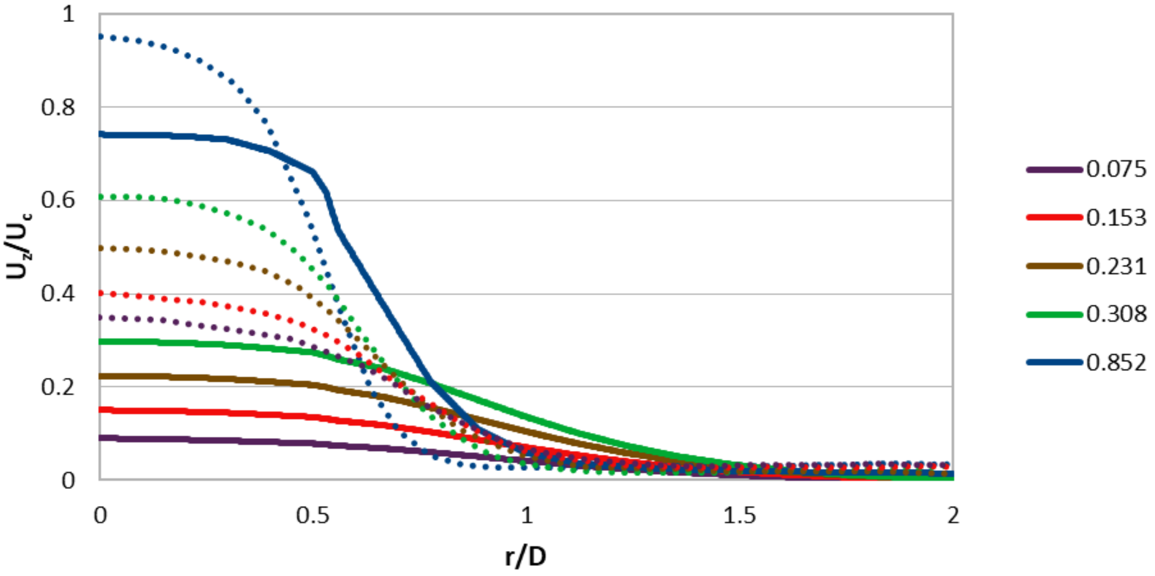


***Figure 29.** Mean axial velocity profiles at different axial distances from the impingement plate for the first mesh and the experiment.*

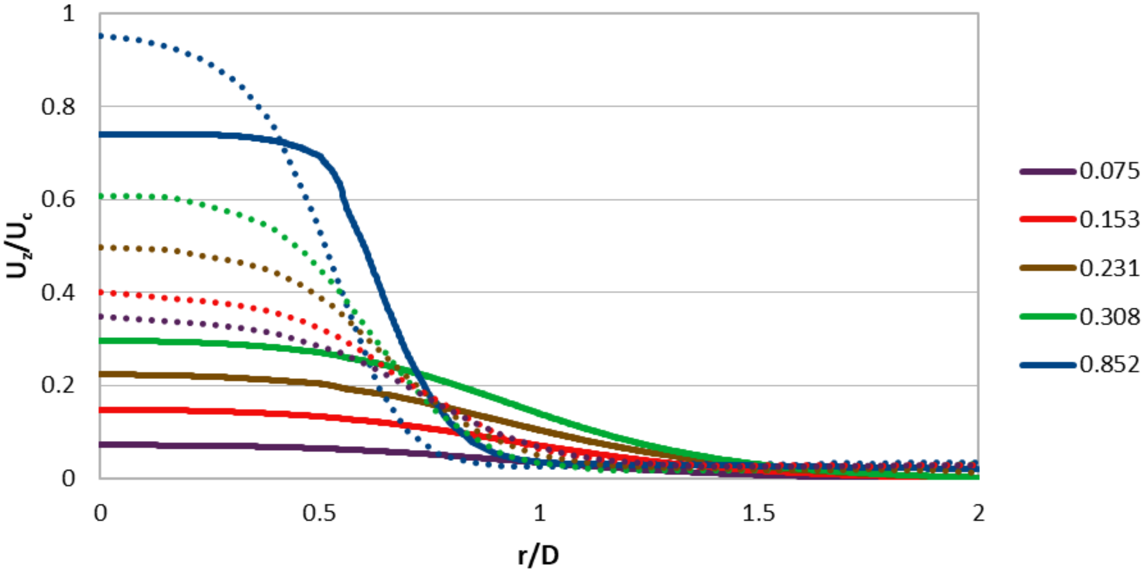


***Figure 30.** Mean axial velocity profiles at different axial distances from the impingement plate for the second mesh and the experiment.*

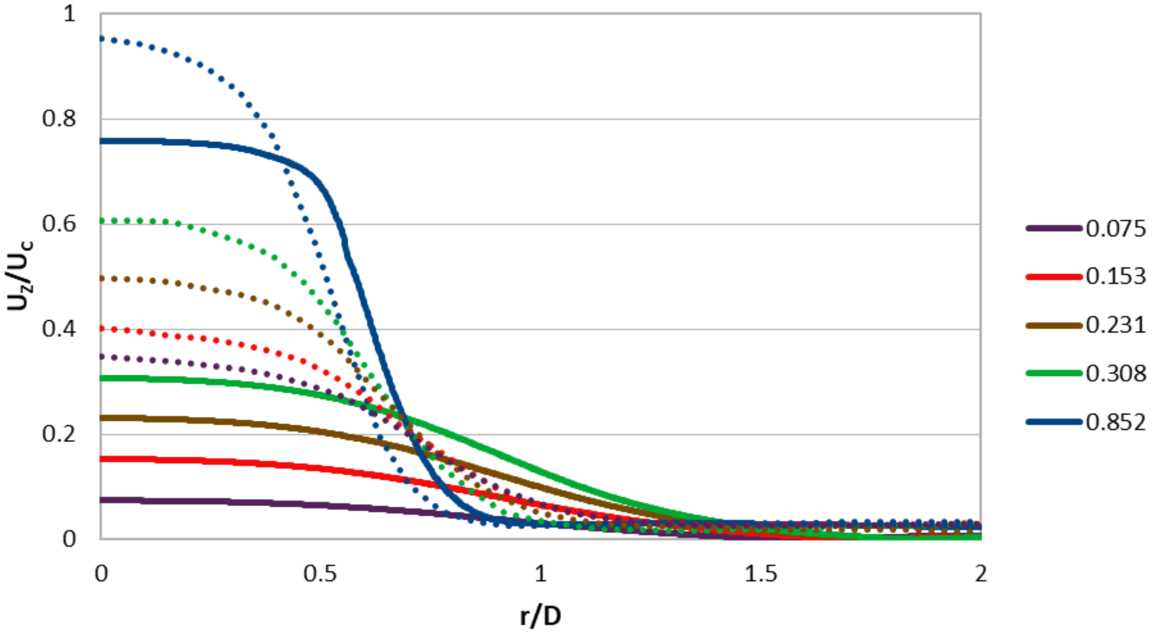


Figure 31. *Mean axial velocity profiles at different axial distances from the impingement plate for the third mesh and the experiment.*

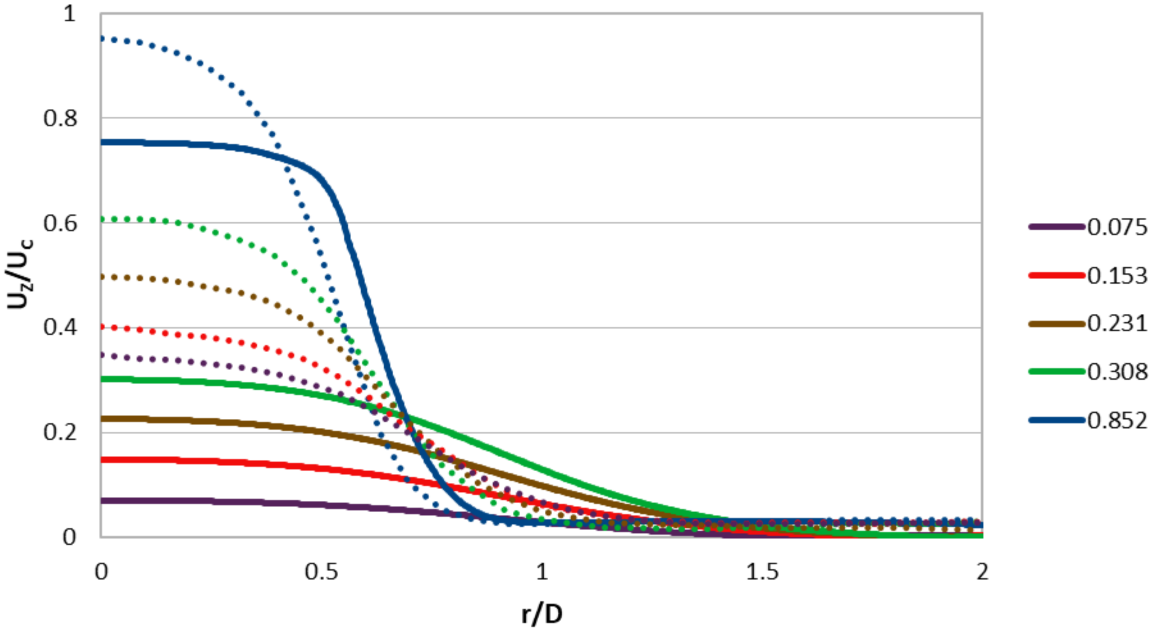


Figure 32. *Mean axial velocity profiles at different axial distances from the impingement plate for the fourth mesh and the experiment.*

Moving on, figures 33, 34, 35, 36 depict the mean radial velocity profiles at different radial locations from the centerline.

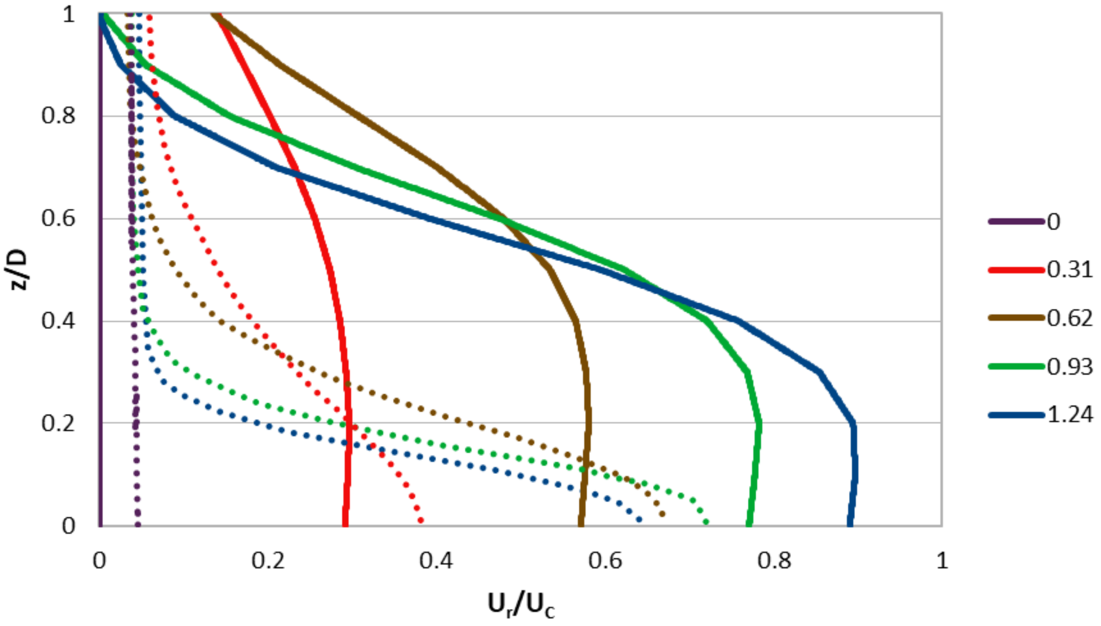


Figure 33. *Mean radial velocity profiles at different radial locations from the centerline for the first mesh and the experiment.*

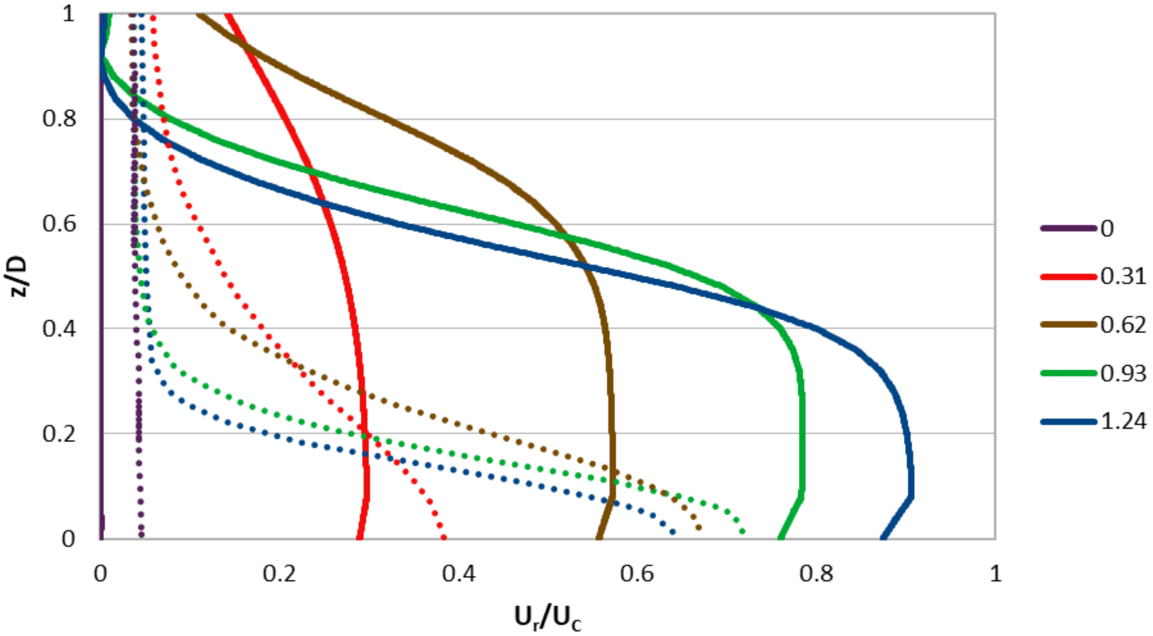


Figure 34. *Mean radial velocity profiles at different radial locations from the centerline for the second mesh and the experiment.*

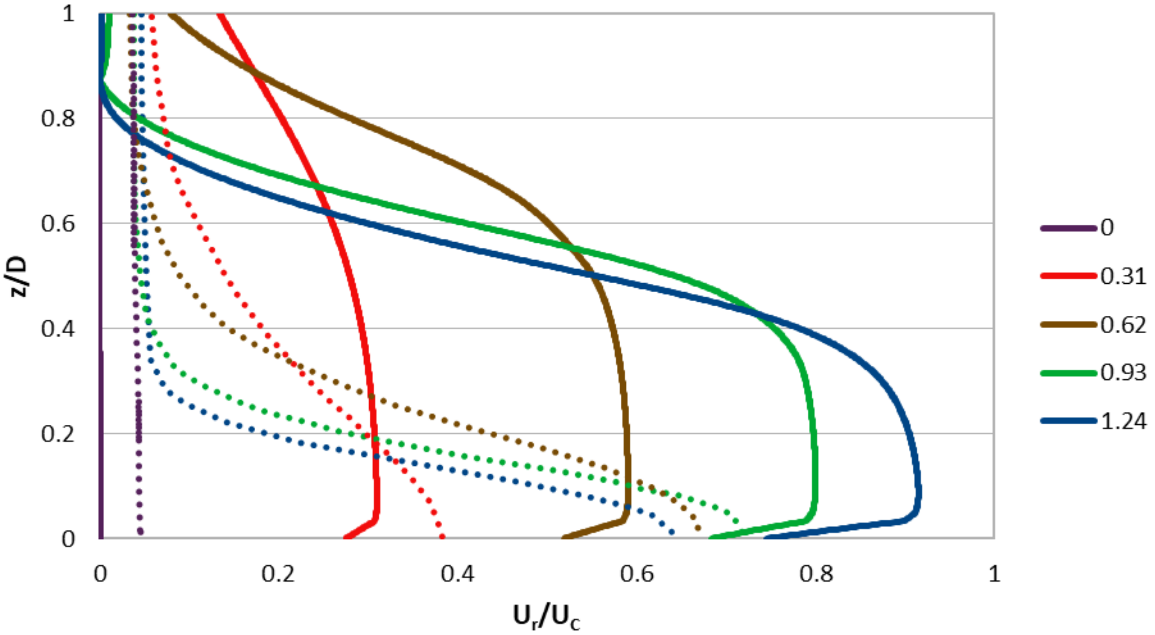


__Figure 35.__ Mean radial velocity profiles at different radial locations from the centerline for the third mesh and the experiment.

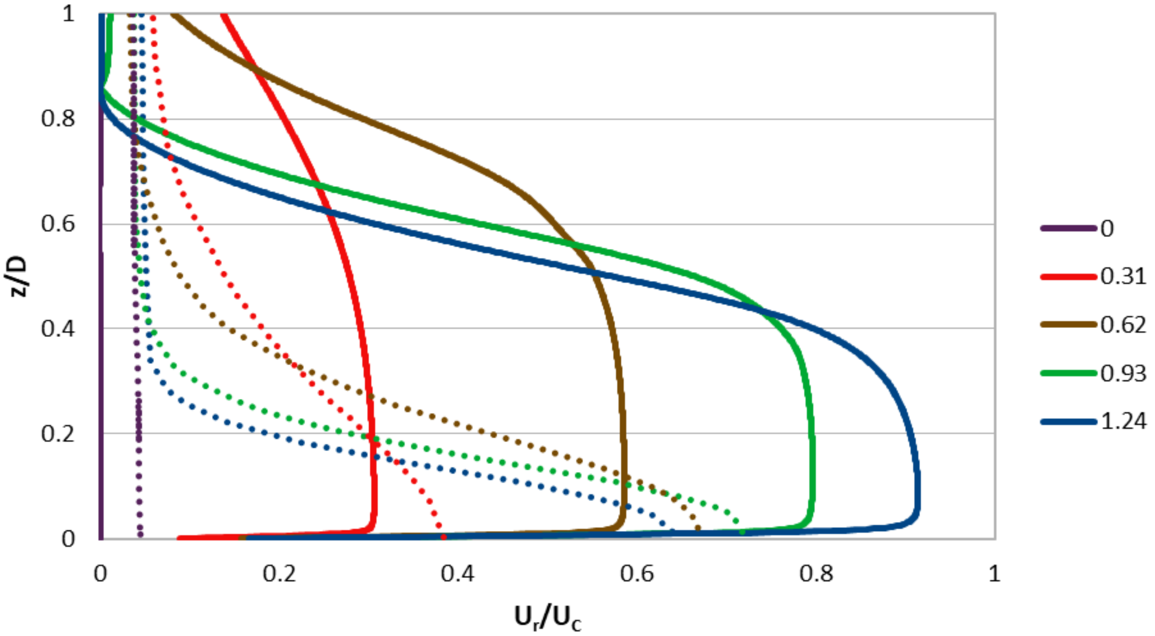


__Figure 36.__ Mean radial velocity profiles at different radial locations from the centerline for the fourth mesh and the experiment.

Finally, the last diagrams of these 2D simulations refer to the mean axial velocity profiles at different radial locations from the centerline. The figures for the four meshes and the experimental results are figures 37, 38, 39 and 40 below.

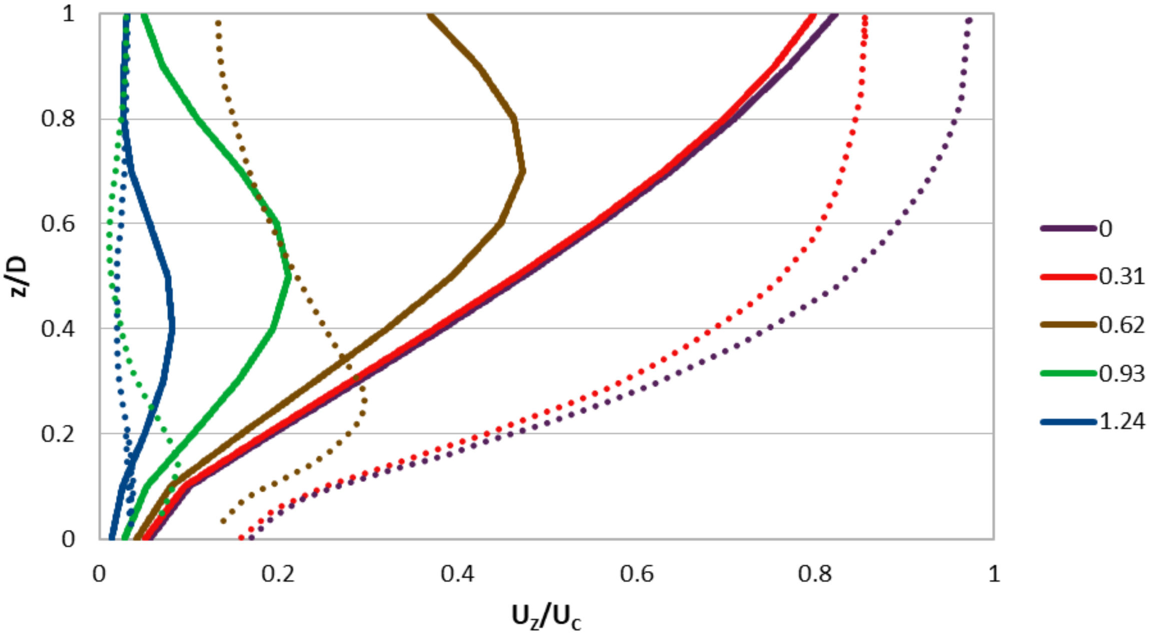


Figure 37. *Mean axial velocity profiles at different radial locations from the centerline for the first mesh and the experiment.*

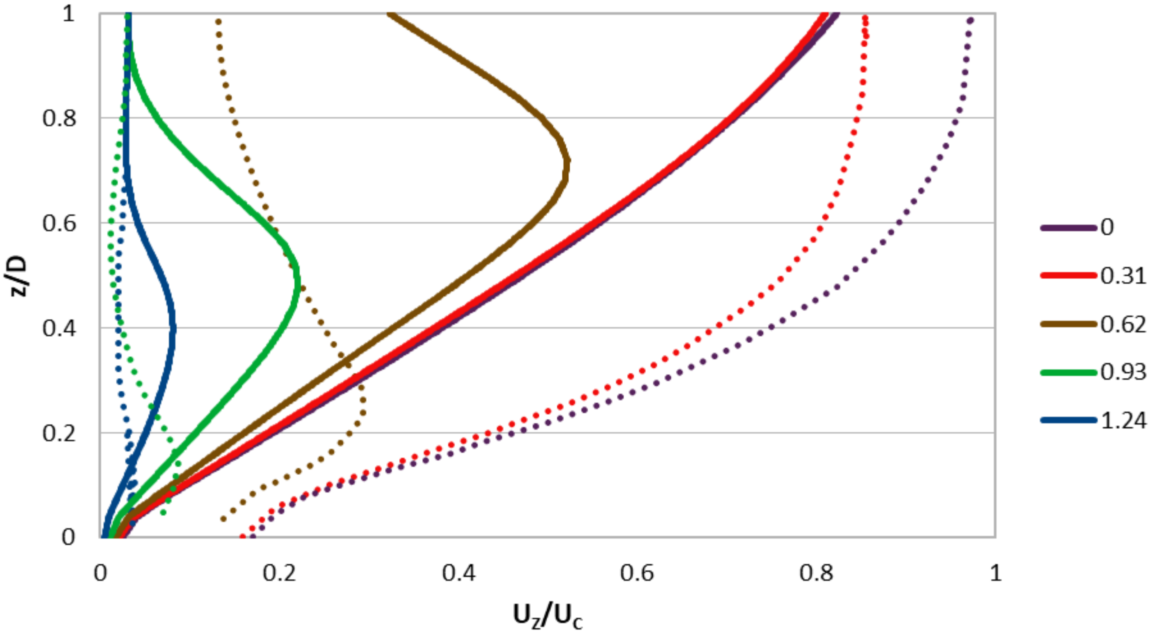


Figure 38. *Mean axial velocity profiles at different radial locations from the centerline for the second mesh and the experiment.*

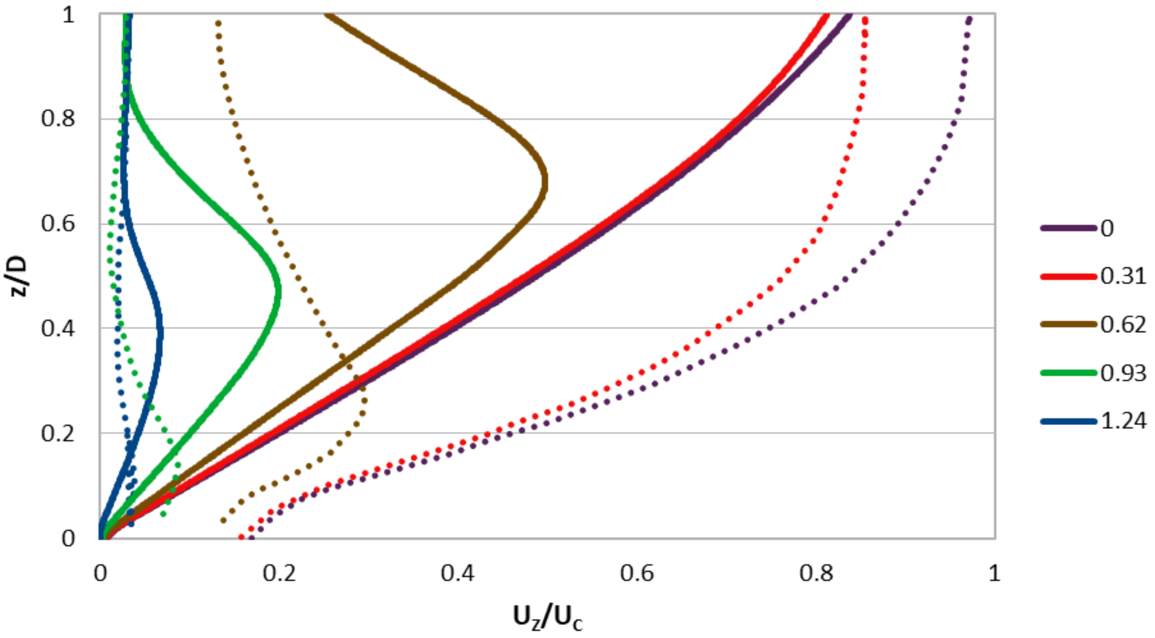


__Figure 39.__ Mean axial velocity profiles at different radial locations from the centerline for the third mesh and the experiment.

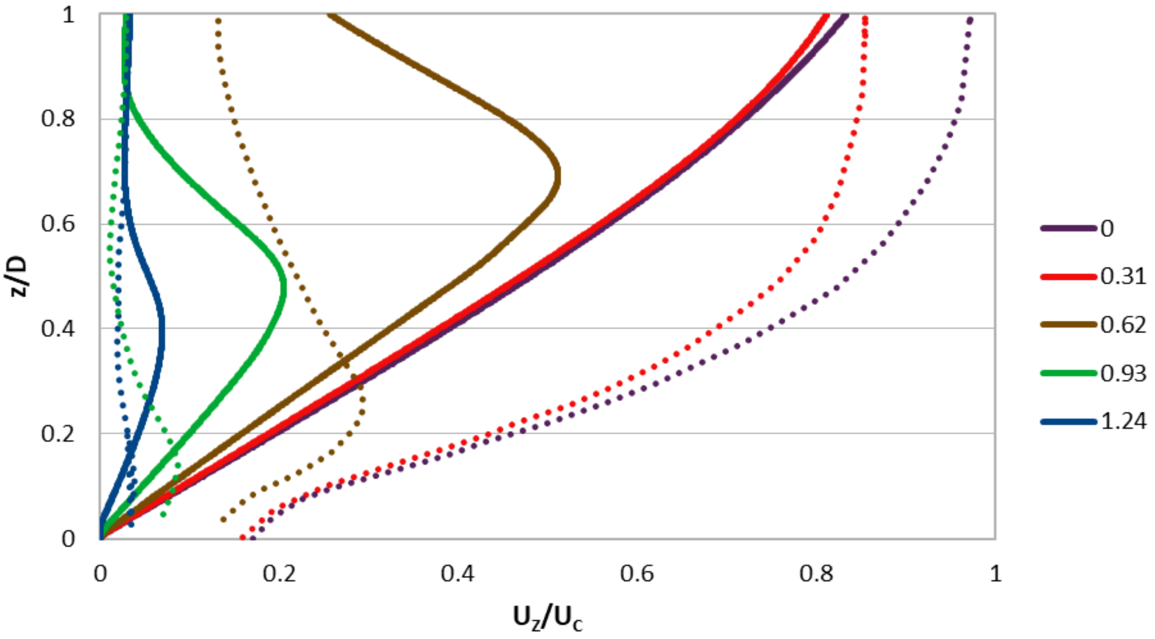


__Figure 40.__ Mean axial velocity profiles at different radial locations from the centerline for the fourth mesh and the experiment.

2.6 Results overview

All in all, from the diagrams it is understandable that the results obtained in these simulations are mesh independent. Despite using different number of cells and different y^+ values for the bottom face, the results for the different locations seem to be very close for all the four meshes. It is visible that the biggest differences in the results are obtained between the two extreme cases: the first and the fourth mesh. Specifically, in figure 23, the maximum deviation in the results between these two meshes is 0.1847 m/s, while the average deviation is 0.0276m/s. For figure 24, which is the only plot that big differences exist between the meshes, the respective values are 0.3609 m/s and 0.2518 m/s. Moving on to figures 25 and 28, and speaking of the blue graphs (z/D=0.852) for which the biggest difference is observed, the respective values are 0.1122 m/s and 0.0464 m/s. For figures 29 and 32 and the same colour graph, the values are 0.1171 m/s and 0.0265 m/s, while for figures 33 and 36 (r/D=1.24), 0.7236 m/s and 0.0481 m/s. Finally, for figures 37 and 40, and speaking of the light blue graphs, the respective values are 0.1847 m/s and 0.0276 m/s. Through the results comparison, it can be understood that apart from figure 24 for which wall-functions don't seem to give similar results with meshes 3 and 4 (low-Reynolds form), all meshes give close results. That also confirms F. Menter's *et al.* finding, who supported that for the k-ω SST model, the whole $0<y^+<300$ range can accurately be applied, with the use of the appropriate wall-functions for the high-Reynolds form. However, mesh 3 seems to be the best choice, combining better results with respect to the experimental ones, and manageable computational time.

Nevertheless, although there is good agreement between the different meshes' results, except for figures 23 and 24 (using the low-Reynolds form of k-ω SST), all other plots indicate weak accuracy of the numerical simulations compared with the experimental findings. Apart from figures 25-28, which indicate the mean radial velocity profiles at different axial distances from the impingement plate, all other diagrams show relatively good qualitative agreement with the experimental results, but not quantitative. As it can be seen from figure 22, as the jet approaches the impingement surface, rapid flow deceleration and shear layer growth are observed, due to the turbulent shear stresses, as they are also observed in the experimental results. The wall-jet region can also be detected. There, a rapid acceleration takes place due to the presence of a stagnation zone, which results in the radial deflection of the jet flow. The strong gradients inside the wall-jet region can also be seen by the velocity contours, together with the flow's deceleration further away radially. As it can be seen from the contour plot of figure 22, the main issue of the numerical simulations is that the accelerating zone inside the wall-jet region is much bigger than it should be, according to the experimental findings. This is the reason why figures 25-32 overpredict velocities at those locations. In the last plots that show velocities at the $0<z/D<1$ region, accuracy is also absent.

2.7 Conclusions

Summarizing, the results obtained via the above simulations conducted by *OpenFOAM*, proved to be mesh independent, and although they were relatively qualitative, they were quantitatively inadequate. That is in good agreement with many authors' belief that 2D simulations are good for qualitative study, but not for quantitative predictions [66]. It is highly likely that the results' inaccuracy comes from the absence of the third dimension. In 2D flows, it is assumed that there is no change of velocity in the third dimension, which means either that the velocity is constant, or zero in this direction. This can't be true in a turbulent flow like in our case, because turbulence is a 3D phenomenon by nature. The main characteristics of turbulence are its irregularity and diffusivity, which means that it has chaotic nature, and it causes increased rates of momentum transfer, in all three dimensions. Consequently, by ignoring the third dimension, the momentum transfer on the one dimension is also ignored, thus leading to false predictions. Also, 2D simulations neglect the so called «vortex stretching», which is the lengthening of vortices in 3D flows that causes an increase in vorticity in the stretching direction [67]. At the same time, it is a mechanism that guarantees the conservation of angular momentum, and this is the reason it happens at first place. Another factor that could have contributed to the inaccuracy of the solutions is the 1st order schemes used for «div(k)», «div(omega)» and «div(R)». The above conclusions, led to the study of a new 3D case in a similar, yet different jet impingement geometry.

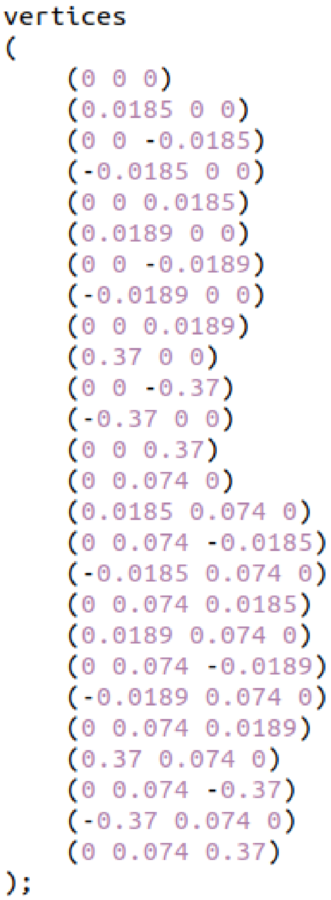
Unit 3. **THREE-DIMENSIONAL SIMULATIONS**

3.1 Introduction

As mentioned above, after the 2D simulations, the investigation of impinging jets in three-dimensions followed. This time, the aim was at the validation and verification of some of the results of Zahir U. Ahmed *et al.* [68], in order to improve the understanding of impinging jet flow characteristics. In their paper, both non-swirling and swirling jets were investigated, although in the thesis' simulations only the former case was considered. In addition, for their numerical simulations, the authors used the commercial software package *FLUENT* (version 14.5), while no extensive description of the experimental procedure was given on the reference paper. In this chapter, a steady-state and a transient simulation is going to be addressed and a comparison with the numerical and experimental findings of the paper is going to be made. In the simulations, similar conditions and settings with the paper's simulations were used, but some differentiations were necessary, due to the differences in the computational domain and the CFD softwares used.

3.2 Geometry of the simulations

Contrary to the geometries of the previous simulations, in these new simulations only a part of the whole geometry was considered. It is about a cylindrical domain with a height equal to the nozzle-to-plate distance, thus ranging from the nozzle exit to the impingement surface. Concerning the radial boundaries, after many tests conducted by the authors, it was found that a radius of 10 nozzle diameters was sufficiently away, so that they did not impact the velocity field. Once again, the geometry was created in the «BlockMeshDict» dictionary file, and the vertices definition follows in figure 41. The geometry is depicted in figure 42 and is visualized in *Paraview* environment. It is made up of five boundary faces: the «inlet», the «outlet», the «top», the «thickness» and the «bottom» faces, also illustrated in figure 42. Walls are the faces: «thickness» and «bottom». Concerning the rest of the geometric lengths, the nozzle diameter equals 37 mm, and its thickness equals 0.4 mm.



```
vertices
(
    (0 0 0)
    (0.0185 0 0)
    (0 0 -0.0185)
    (-0.0185 0 0)
    (0 0 0.0185)
    (0.0189 0 0)
    (0 0 -0.0189)
    (-0.0189 0 0)
    (0 0 0.0189)
    (0.37 0 0)
    (0 0 -0.37)
    (-0.37 0 0)
    (0 0 0.37)
    (0 0.074 0)
    (0.0185 0.074 0)
    (0 0.074 -0.0185)
    (-0.0185 0.074 0)
    (0 0.074 0.0185)
    (0.0189 0.074 0)
    (0 0.074 -0.0189)
    (-0.0189 0.074 0)
    (0 0.074 0.0189)
    (0.37 0.074 0)
    (0 0.074 -0.37)
    (-0.37 0.074 0)
    (0 0.074 0.37)
);
```

Figure 41. *A part of the BlockMeshDict, in which the geometry is defined via vertices.*

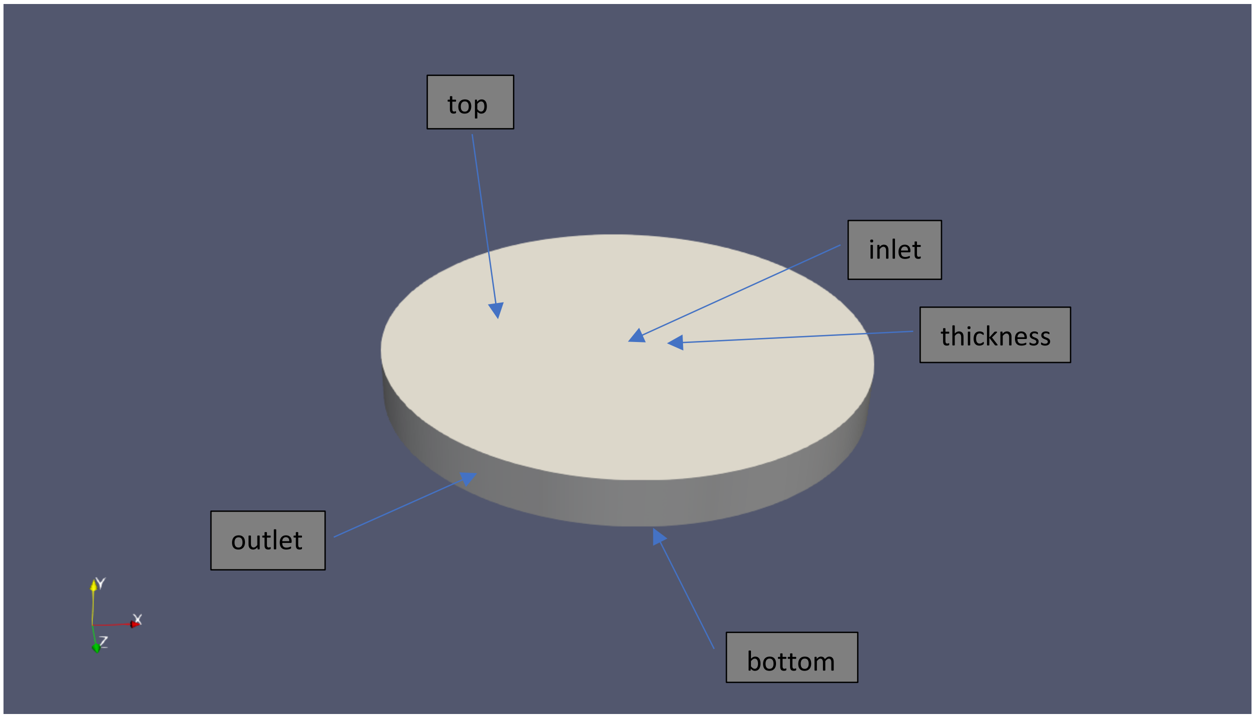


Figure 42. *Geometry with boundary faces.*

3.3 Simulations' parameters

The different flow properties are concentrated in table 6. It must be clarified that the working fluid is not specified in the reference paper, yet the kinematic viscosity was calculated via the right-hand side of *(1.34)*.

Table 6. *Flow properties [68].*

Bulk velocity	9.6 m/s
Reynolds number	23,000
Kinematic viscosity	1.5443x10^{-5} m^2/s
Turbulence intensity at inlet	2%
Turbulence intensity at outlet	5%

3.4 Mesh

The mesh was once again generated by the «blockMesh» command at Ubuntu's terminal, consisting of 14,528,000 hexahedra cells. Again, the geometry was divided into blocks with different grading in each one, and the mesh thickened the closer the distance got with respect to the impinging target and coarsened radially. This way, the drastic changes that the flow is subject to near the region of interest, and the steep gradients there, could be taken into account successfully. «checkMesh» showed no issues with the quality of the mesh. Its top view and the facade can be seen in figures 43, 44. Concerning the turbulence model used (Re=23,000), that was the k-ω SST model and specifically its low-Reynolds form, as the y$^+$ value approached the value of 2.30.

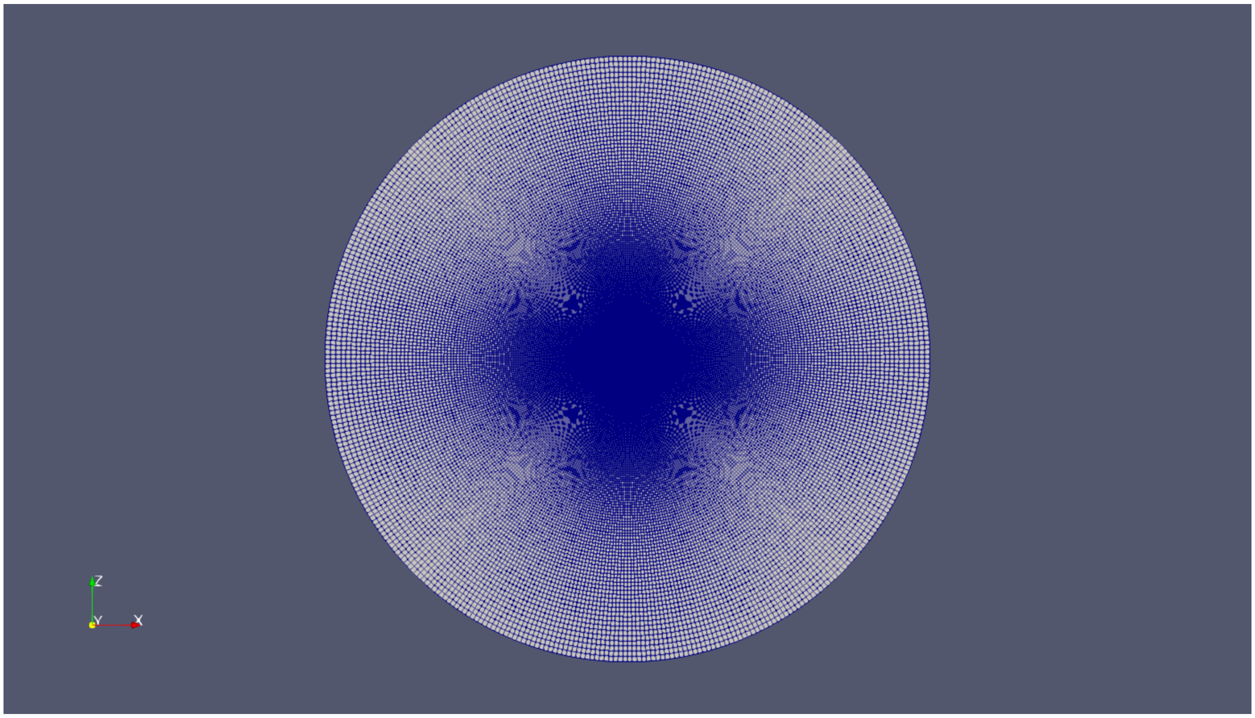


Figure 43. *The top view of the mesh.*

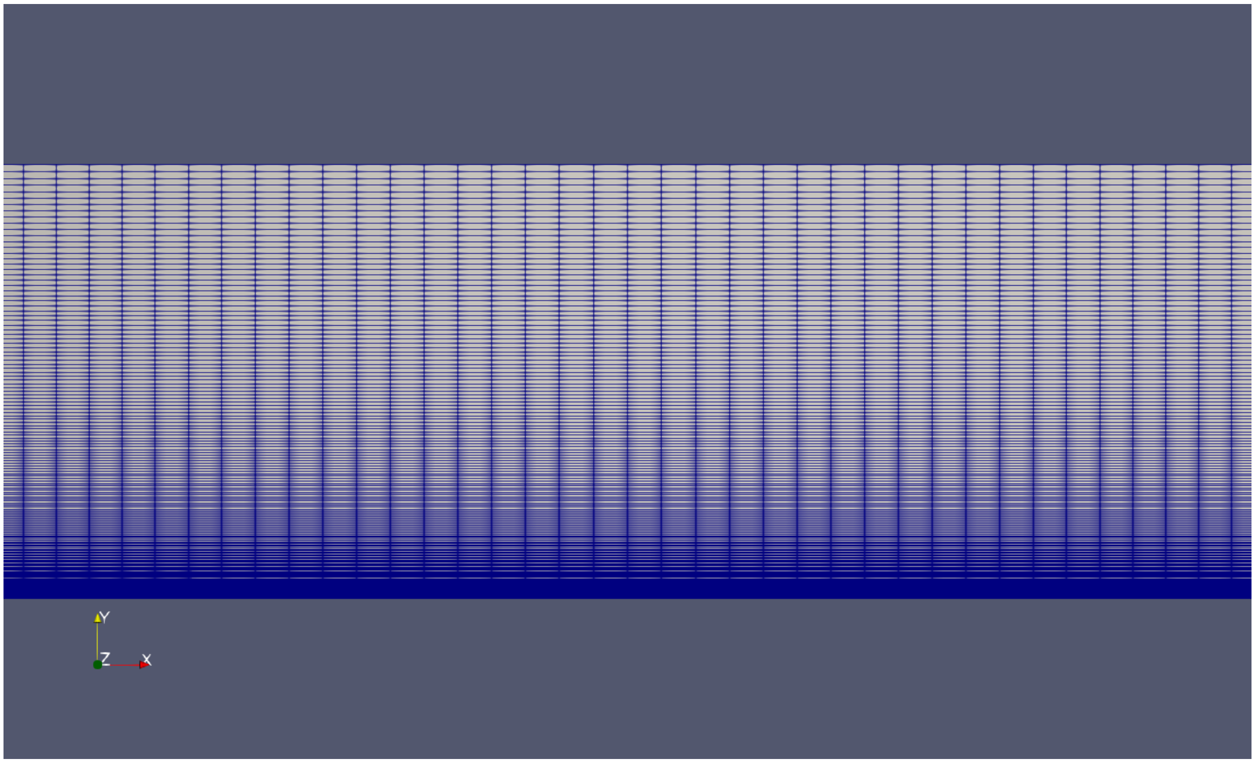


Figure 44. *A part of the facade of the mesh.*

3.5 The steady-state case

For the definition of the turbulent variables, once again equations *(1.52)-(1.56)* were used. The initial conditions for the steady-state simulations are gathered in table 7, below.

Table 7. *Initial conditions.*

U (m^2/s)	p (m^2/s^2)	k (m^2/s^2)	omega (1/s)	nut (m^2/s)
(0 0 0)	0	0.06	174.2169	0.000344

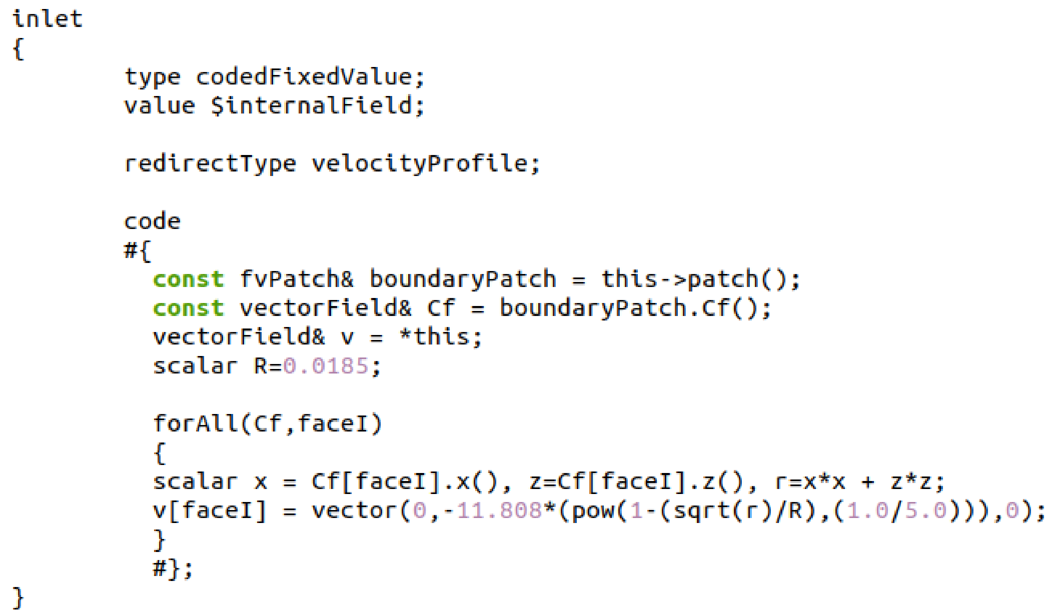
The boundary conditions follow in table 8. It must be mentioned that for the inlet, a velocity profile was applied, which obeyed the equation:

$$U_{in} \,/\, U_b = 1.23 \cdot (1 - r \,/\, R)^{1/5}. \tag{3.1}$$

This profile was applied via a code written in *C++* language, shown in figure 45. Moreover, due to the specification of the turbulence intensity at both the inlet and the outlet, two boundary conditions concerning it are applied, for k and omega variables. The type of the boundary condition used for k in *OpenFOAM* is called «turbulentIntensityKineticEnergyInlet», and it is applied for both the inlet and the outlet. For this condition, the value of intensity must also be specified, taking the value of 0.02 for the inlet and 0.05 for the outlet. It sets the turbulent kinetic energy based on the patch velocity and user-supplied turbulence intensity. For omega, the respective condition is called «turbulentMixingLengthFrequencyInlet», and the value of the mixing length should be defined in this case. It sets the turbulent specific dissipation rate based on the patch turbulence kinetic energy and user-supplied mixing length. The mixing length obeys the expression:

Mixing length= $0.07D_h$. *(3.2)*

More details about these two types of boundary conditions can be found in literature [69].



```
inlet
{
        type codedFixedValue;
        value $internalField;

        redirectType velocityProfile;

        code
        #{
          const fvPatch& boundaryPatch = this->patch();
          const vectorField& Cf = boundaryPatch.Cf();
          vectorField& v = *this;
          scalar R=0.0185;

          forAll(Cf,faceI)
          {
          scalar x = Cf[faceI].x(), z=Cf[faceI].z(), r=x*x + z*z;
          v[faceI] = vector(0,-11.808*(pow(1-(sqrt(r)/R),(1.0/5.0))),0);
          }
          #};
}
```

Figure 45. *The «codedFixedValue» boundary condition for the inlet velocity profile.*

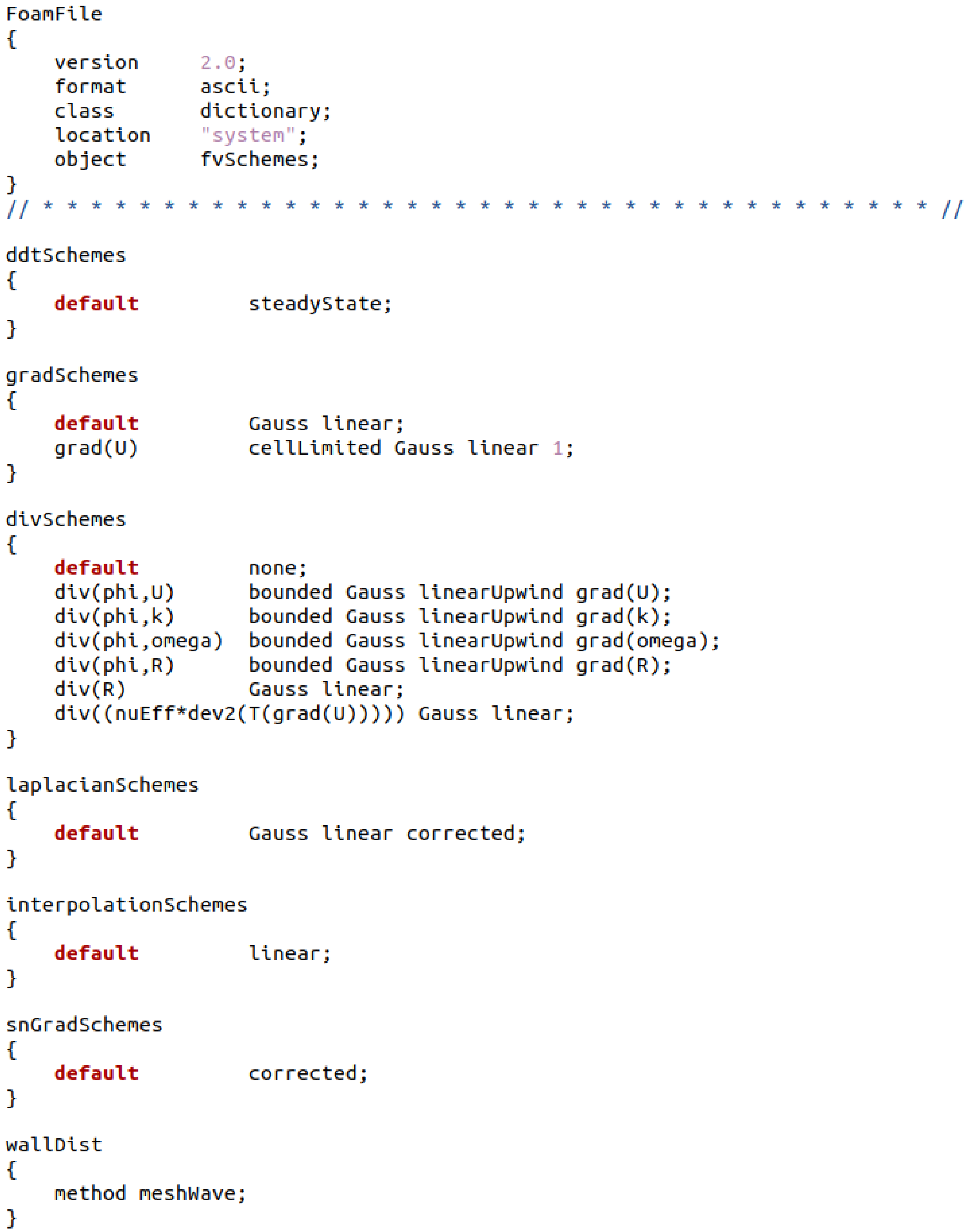
Table 8. *Boundary conditions for the steady-state simulation.*

	Inlet	Outlet	Top	Thickness	Bottom
U (m^2/s)	codedFixed Value	zeroGradient	fixedValue (0 0 0)	noSlip	noSlip
p (m^2/s^2)	zeroGradie nt	fixedValue 0	zeroGradient	zeroGradient	zeroGradient
k (m^2/s^2)	value 0.06	value 0.06	fixedValue 0	fixedValue 1e-11	fixedValue 1e-11
omega (1/s)	value 174.2169	zeroGradient	zeroGradient	omegaWallFunctio n 174.2169	omegaWallFunction 174.2169
nut (m^2/s)	fixedValue 0.000344	fixedValue 0.000344	fixedValue 0.000344	fixedValue 0.000344	fixedValue 0.000344

The «noSlip» boundary condition is same as the «fixedValue (0 0 0)», appropriate for the contact points between the fluid and the walls. The value of 1e-11 was assigned for k on the wall faces, as an alternative way of assigning the zero value, leading to better results.

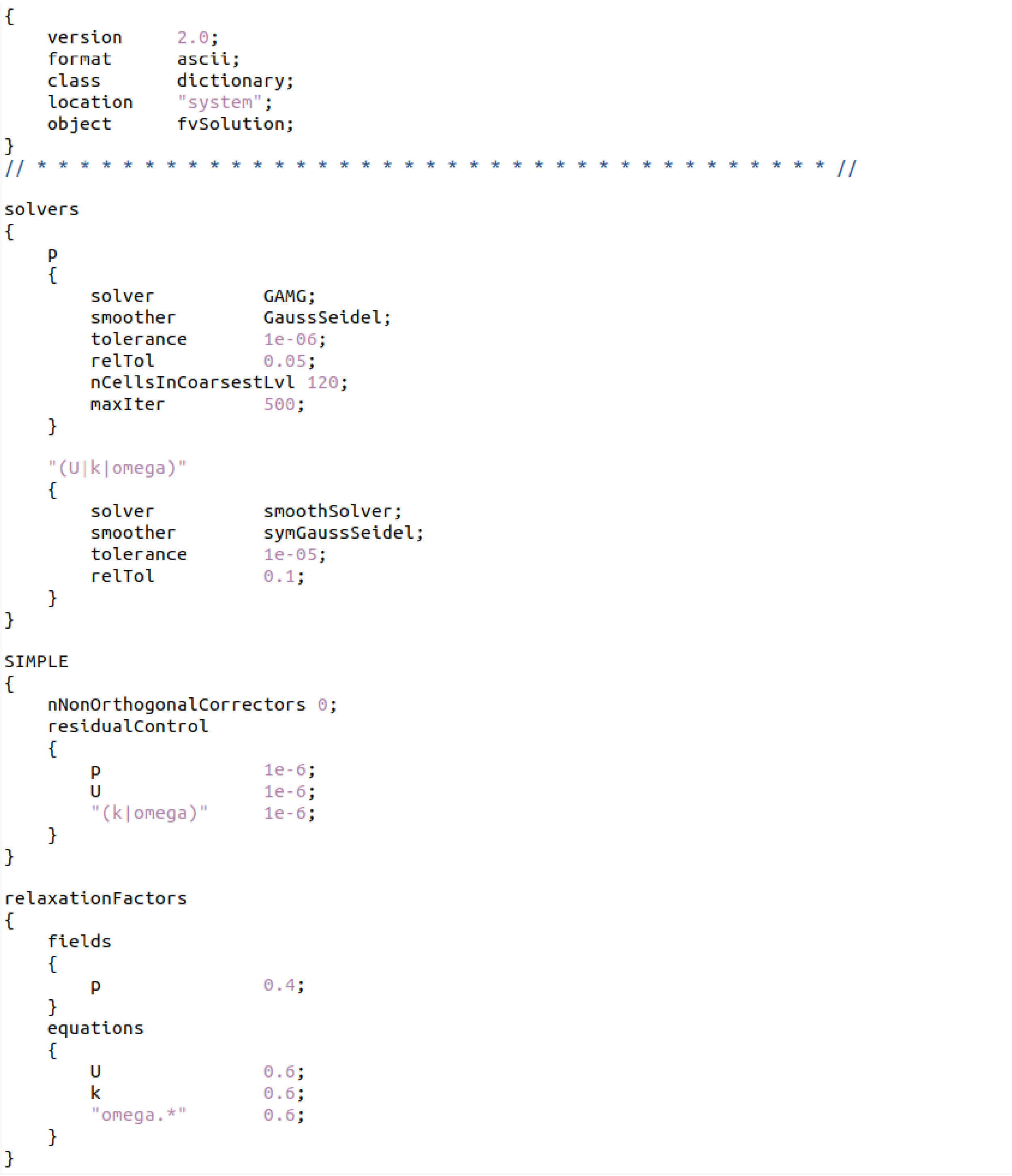
Moving on, the discretization schemes of the simulations are shown in figure 46. This time, unlike the 2D simulations, all schemes were 2nd order, ensuring the accuracy of the solution. For the «grad(U)» term, «cellLimited Gauss linear 1» was used to offer stability to the solution. The purpose of gradient limiters is to avoid over and under shoots on the gradient computations.

Figure 47 shows the «fvSolutions» directory of the simulation. In comparison with the 2D simulations, some changes in the values of the settings were made and some additional terms were added. «maxIter» stops the simulation when the number of the variable’s iterations reaches its value, i.e., in our case the value of 500. «nCellsInCoarsestLevel» is the mesh’s number of cells at the coarsest level. The generalised method of geometric-algebraic multi-grid (GAMG) uses the principle of generating a quick solution on a very coarse mesh and afterwards, mapping the solution onto a finer mesh: using it as an initial guess for the fine mesh. As a result, «nCellsInCoarsestLevel» refers to the initial mesh. For the simulation, 15,000 iterations were carried out. Before moving on to the presentation of the results, the description of the transient simulation is going to be done, in order to compare them between the two simulations.



```
FoamFile
{
    version     2.0;
    format      ascii;
    class       dictionary;
    location    "system";
    object      fvSchemes;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //

ddtSchemes
{
    default         steadyState;
}

gradSchemes
{
    default         Gauss linear;
    grad(U)         cellLimited Gauss linear 1;
}

divSchemes
{
    default         none;
    div(phi,U)      bounded Gauss linearUpwind grad(U);
    div(phi,k)      bounded Gauss linearUpwind grad(k);
    div(phi,omega)  bounded Gauss linearUpwind grad(omega);
    div(phi,R)      bounded Gauss linearUpwind grad(R);
    div(R)          Gauss linear;
    div((nuEff*dev2(T(grad(U))))) Gauss linear;
}

laplacianSchemes
{
    default         Gauss linear corrected;
}

interpolationSchemes
{
    default         linear;
}

snGradSchemes
{
    default         corrected;
}

wallDist
{
    method meshWave;
}
```

Figure 46. *Discretization schemes for the steady-state 3D simulation in the OpenFOAM fvSchemes dictionary.*



```
{
    version         2.0;
    format          ascii;
    class           dictionary;
    location        "system";
    object          fvSolution;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //

solvers
{
    p
    {
        solver          GAMG;
        smoother        GaussSeidel;
        tolerance       1e-06;
        relTol          0.05;
        nCellsInCoarsestLvl 120;
        maxIter         500;
    }

    "(U|k|omega)"
    {
        solver          smoothSolver;
        smoother        symGaussSeidel;
        tolerance       1e-05;
        relTol          0.1;
    }
}

SIMPLE
{
    nNonOrthogonalCorrectors 0;
    residualControl
    {
        p               1e-6;
        U               1e-6;
        "(k|omega)"     1e-6;
    }
}

relaxationFactors
{
    fields
    {
        p               0.4;
    }
    equations
    {
        U               0.6;
        k               0.6;
        "omega.*"       0.6;
    }
}
```

Figure 47. *Solution and algorithm control for the 3D steady-state simulation in the OpenFOAM fvSolutions dictionary.*

3.6 The transient case

The transient simulation was carried out as a test to whether or not the unsteady nature of the problem could give more accurate results, with respect to the experimental ones. In addition, it was a test as to whether or not it could be part of a CFDEM (Computational Fluid Dynamics coupled with Discrete Element Method simulations) simulation conducted in a future thesis, in which particles would enter from the inlet and erosion would be studied on the bottom face. The differentiation to a transient case, means the introduction of time as an additional variable. Apparently, the geometry, the mesh, the simulation parameters, and the boundary conditions remained untouched. As initial conditions, the results of the last iteration of the steady-state simulation were used, so as to increase the stability of the solution.

By adding the time variable, an additional discretization scheme should be added to account for the discretization of time. For that purpose, the «CrankNicolson» time scheme was introduced, which is a 2nd order bounded/unbounded scheme, for which details exist in literature [70]. The blending factor «0.7» is a factor that helps bringing off both accuracy and stability. Specifically, the «0» value corresponds to the pure Euler scheme, which is 1st order bounded, thus offers stability, while «1», corresponds to the pure Crank-Nicolson scheme, which is 2nd order bounded/unbounded, thus offering accuracy. The «fvSchemes» dictionary follows in figure 48.

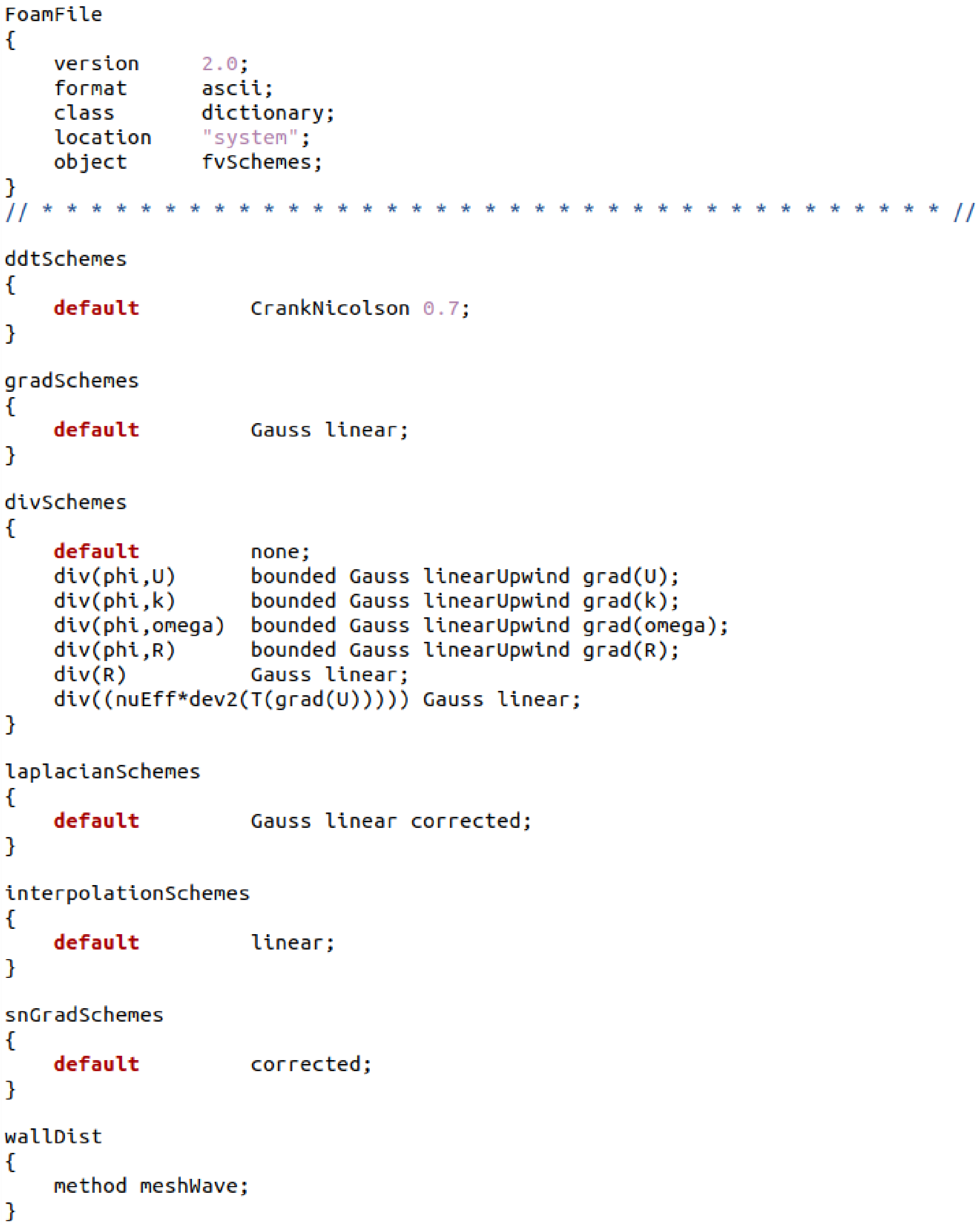
Also, the incompressible solver changed to the «pisoFoam», which uses the Pressure Implicit with Splitting of Operators algorithm (PISO) [71]. This algorithm uses time-steps instead of iterations that the SIMPLE algorithm does, and once again solves for the Navier-Stokes equations. For the «fvSolutions» shown in figure 49, values for the tolerances and the relative tolerances, as well as the solvers and the preconditioners, are those recommended from authors [72].The non-orthogonal correctors were used, in order to account for any non-orthogonality of the mesh, acting only as a benefit for the quality of the results. Finally, the «nCorrectors» were added to improve both the accuracy and the stability.

The only setting left to be determined is the time-step. The time-step chosen for the simulation had to obey a specific condition, which related to the Courant Number (Co). This dimensionless number is expressed as:

$$Co = \frac{\Delta t \cdot |u|}{\Delta x} \tag{3.3}$$

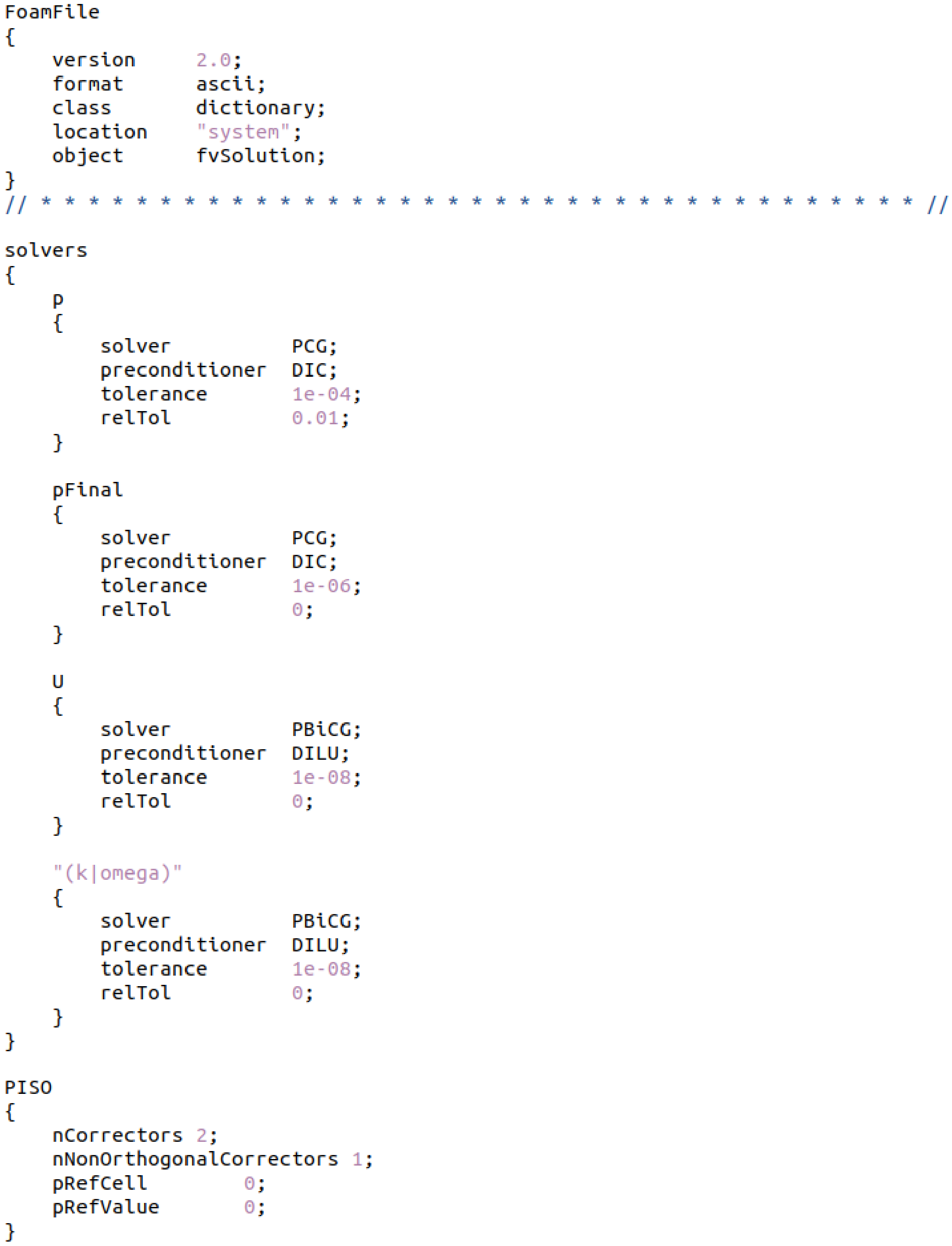
where, Δt: the time-step, $|u|$: the velocity magnitude, Δx: the cell length in the direction of the flow.

For stability reasons, it is known from literature that this number should follow the rule $Co_{max}<1$, for every CFD time-step [73]. Physically, Co expresses the distance any information travels, during a time-step. As a result, the inequality states that this distance should be lower than the cell's length. Faster spread of the information can lead to negligence of different flow characteristics and fluctuations, enlarging errors, thus decreasing solution's stability. In our case, in order to conclude to the largest permissible time-step, ensuring stability and the least computational effort at the same time, Co was assigned the value of Co=1. The minimum cells' length of the domain was approximately l_{min}=0.00002 m and the maximum velocity in these cells was approximately $|u|_{max}$=8 m/s. Consequently, it was found that the time-step should not exceed the value of 0.0000025 s. In the «controlDict» dictionary of the «system» folder, Δt=0.000001 s was assigned. Concerning the run time of the simulation, its value was chosen so as to equal the time needed for the jet to travel 4-5 nozzle-to-plate distances. This time was found to be equal to t=0.03 s.



```
FoamFile
{
    version     2.0;
    format      ascii;
    class       dictionary;
    location    "system";
    object      fvSchemes;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //

ddtSchemes
{
    default         CrankNicolson 0.7;
}

gradSchemes
{
    default         Gauss linear;
}

divSchemes
{
    default         none;
    div(phi,U)      bounded Gauss linearUpwind grad(U);
    div(phi,k)      bounded Gauss linearUpwind grad(k);
    div(phi,omega)  bounded Gauss linearUpwind grad(omega);
    div(phi,R)      bounded Gauss linearUpwind grad(R);
    div(R)          Gauss linear;
    div((nuEff*dev2(T(grad(U))))) Gauss linear;
}

laplacianSchemes
{
    default         Gauss linear corrected;
}

interpolationSchemes
{
    default         linear;
}

snGradSchemes
{
    default         corrected;
}

wallDist
{
    method meshWave;
}
```

Figure 48. *Discretization schemes for the transient 3D simulation in the OpenFOAM fvSchemes dictionary.*



```
FoamFile
{
    version     2.0;
    format      ascii;
    class       dictionary;
    location    "system";
    object      fvSolution;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //

solvers
{
    p
    {
        solver          PCG;
        preconditioner  DIC;
        tolerance       1e-04;
        relTol          0.01;
    }

    pFinal
    {
        solver          PCG;
        preconditioner  DIC;
        tolerance       1e-06;
        relTol          0;
    }

    U
    {
        solver          PBiCG;
        preconditioner  DILU;
        tolerance       1e-08;
        relTol          0;
    }

    "(k|omega)"
    {
        solver          PBiCG;
        preconditioner  DILU;
        tolerance       1e-08;
        relTol          0;
    }
}

PISO
{
    nCorrectors 2;
    nNonOrthogonalCorrectors 1;
    pRefCell        0;
    pRefValue       0;
}
```

Figure 49. *Solution and algorithm control for the 3D transient simulation in the OpenFOAM fvSolutions dictionary.*

3.7 Results of the steady-state and the transient simulations

After the 15,000 iterations for the steady-state simulation and the 0.03s of the transient simulation, residuals for the different variables were investigated. Again, not all residuals reached the value of 1e-5 for the solver to stop, but for both simulations, the fact that their course dropped, afterwards remained constant and their values were sufficiently low, led us to consider solution convergence. Concerning their values, in the steady-state simulation pressure reached the value of 1e-3, omega the value of 1e-5 and the other variables were close to 1e-4. In the transient simulation, pressure reached the value of 1e-4, and the other variables varied between 1e-5 and 1e-7. The results were then exported from *Paraview* to *LibreOffice*, in which the plots were created. This time, apart from the axial and radial velocities at different locations, the wall shear stress and the Reynolds stresses were also detected. *Engauge* was again used for the illustration of the experimental and the numerical results derived from the reference paper. The coordinate system that the results are based, has its origin on the center of the inlet face.

Figure 50 shows the contours of the flow field at a section crossing the coordinate system's origin, in order to have a general perspective of it, being almost identical for both simulations carried out. Thereinafter, figure 51 monitors the Co_{max} during the transient simulation's time-steps, with the illustration carried out with the use of *Gnuplot*, a portable command-line driven graphing utility [74]. Figure 52 depicts a comparison between the experimental data and the authors' and this thesis' numerical findings for the wall shear stress divided by U_b^2, to make it dimensionless. In this case, for the comparison, the case of first cell height equaling Δy=0.05 mm was chosen for the verification, because it equals the value of our's. In figures 53, 54, 55 the comparison for the mean axial-velocity profiles follows, while figures 56, 57 and 58 include the mean radial-velocity profiles, at the same radial locations. Next, figures 59, 60, 61 contain the Reynolds normal stresses, while figures 62, 63, 64 contain the Reynolds shear stresses at the same axial locations. Velocities were divided by the bulk velocity, while Reynolds stresses by U_b^2. For the transient simulations, the results were obtained by time-averaging the results of the various time-steps. However, not all time-steps were averaged, because the results practically remained unchanged with time, so a lot of time was saved without affecting the outcome.

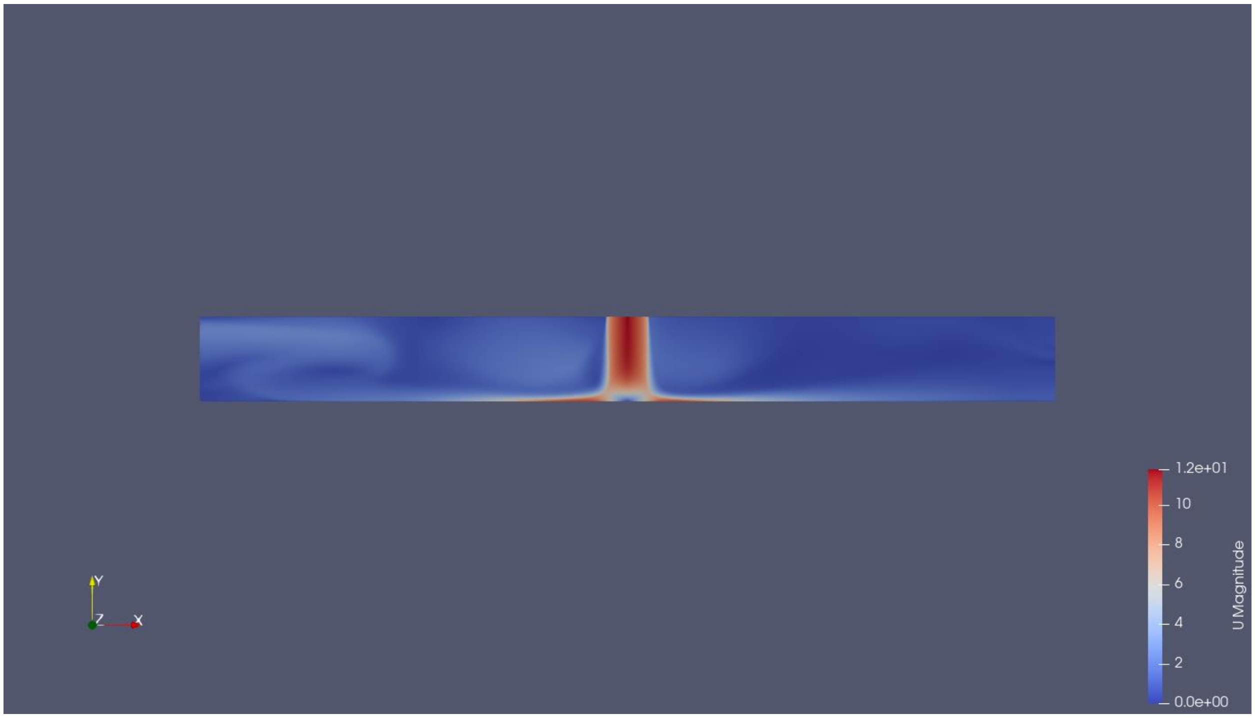


Figure 50. *Section of the 3D steady-state flow field of the simulations in Paraview environment.*

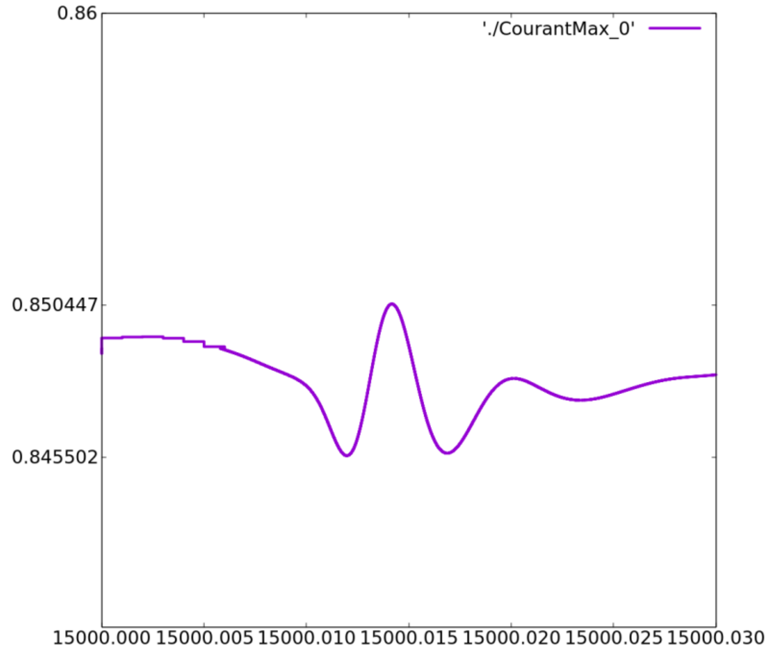


Figure 51. *Maximum Courant number on each time-step of the transient simulation in Gnuplot environment.*

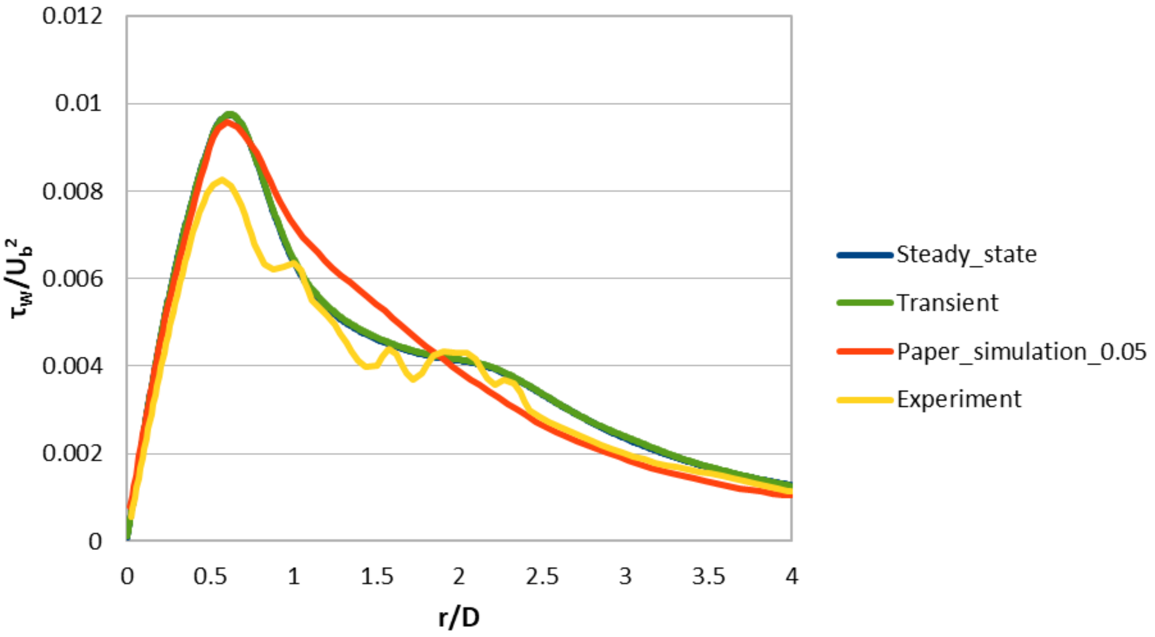


Figure 52. *Comparison of the wall shear stress profile between the experimental data, the numerical simulations of the authors and the thesis' numerical simulations.*

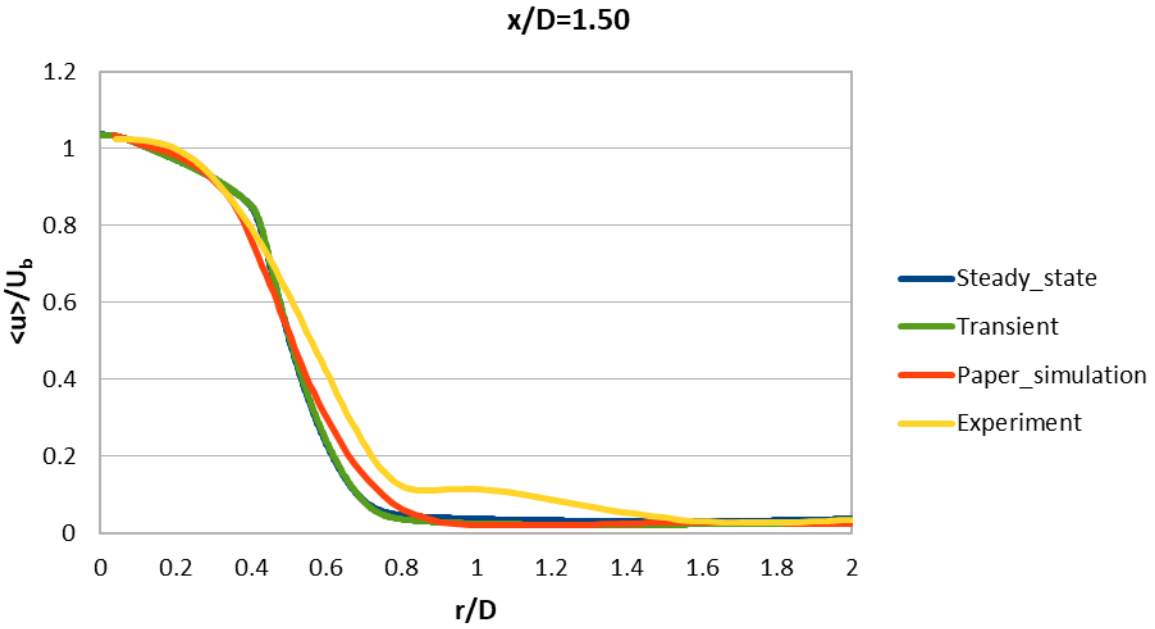


Figure 53. *Comparison of the mean axial velocity profile at x/D=1.50, between the experimental data, the numerical simulations of the authors and the thesis' numerical simulations.*

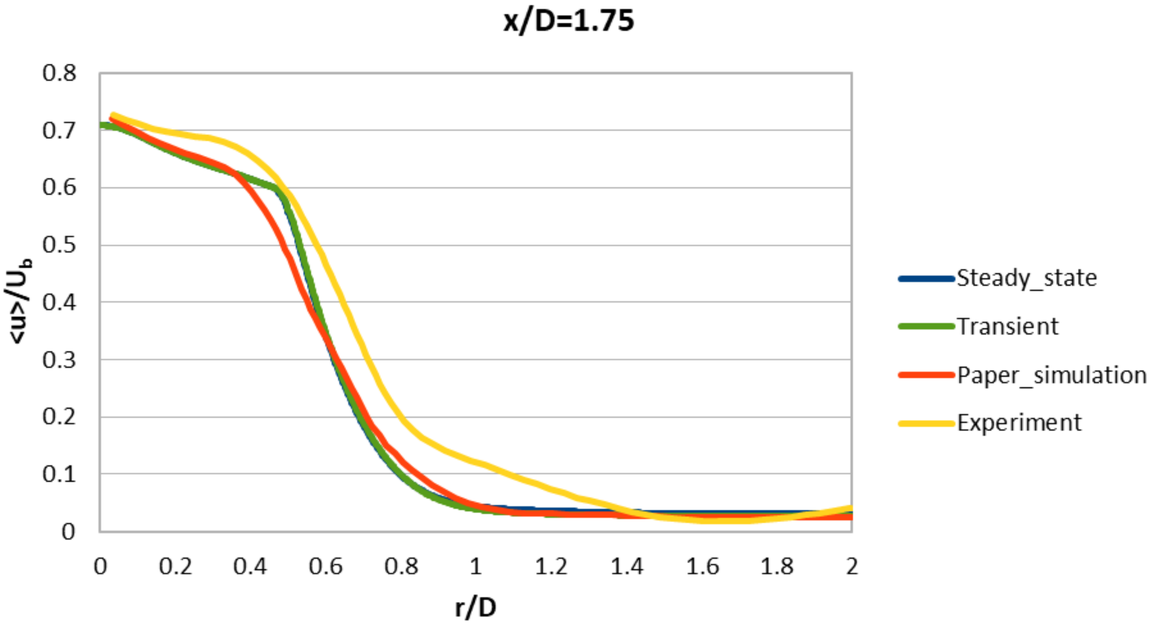


Figure 54. *Comparison of the mean axial velocity profile at x/D=1.75, between the experimental data, the numerical simulations of the authors and the thesis' numerical simulations.*

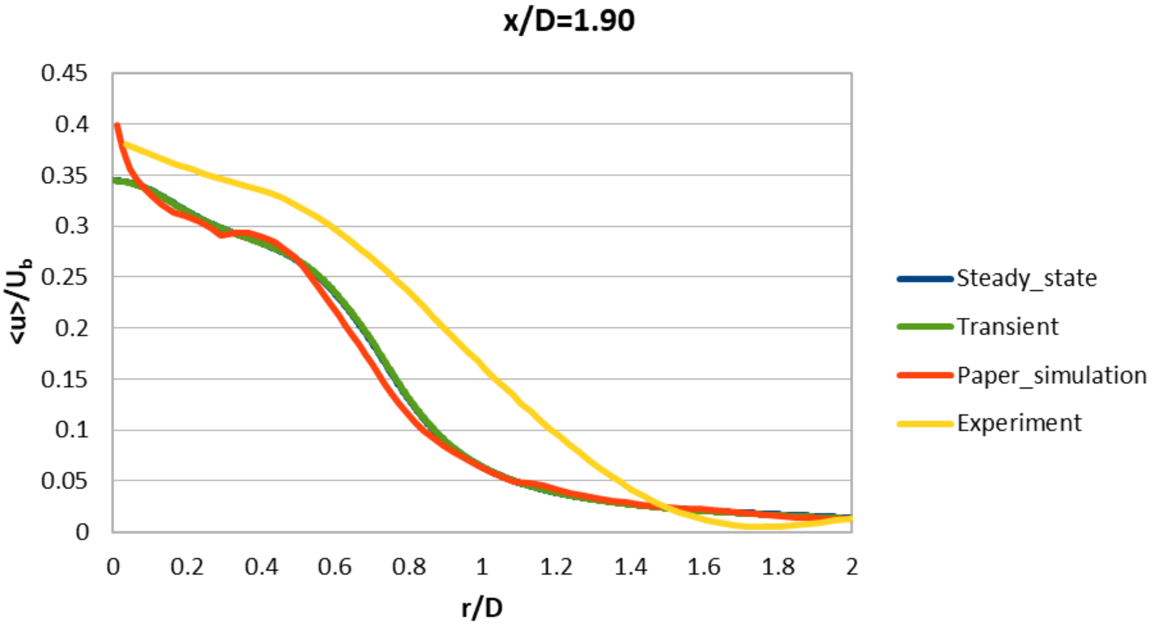


Figure 55. *Comparison of the mean axial velocity profile at x/D=1.90, between the experimental data, the numerical simulations of the authors and the thesis' numerical simulations.*

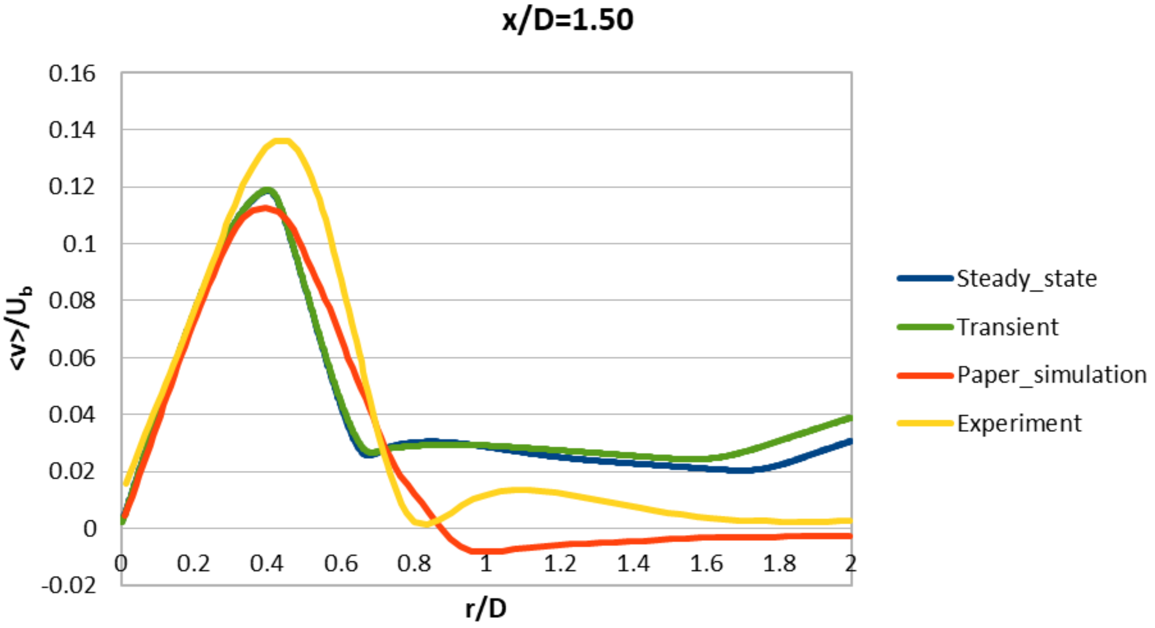


Figure 56. *Comparison of the mean radial velocity profile at x/D=1.50, between the experimental data, the numerical simulations of the authors and the thesis' numerical simulations.*

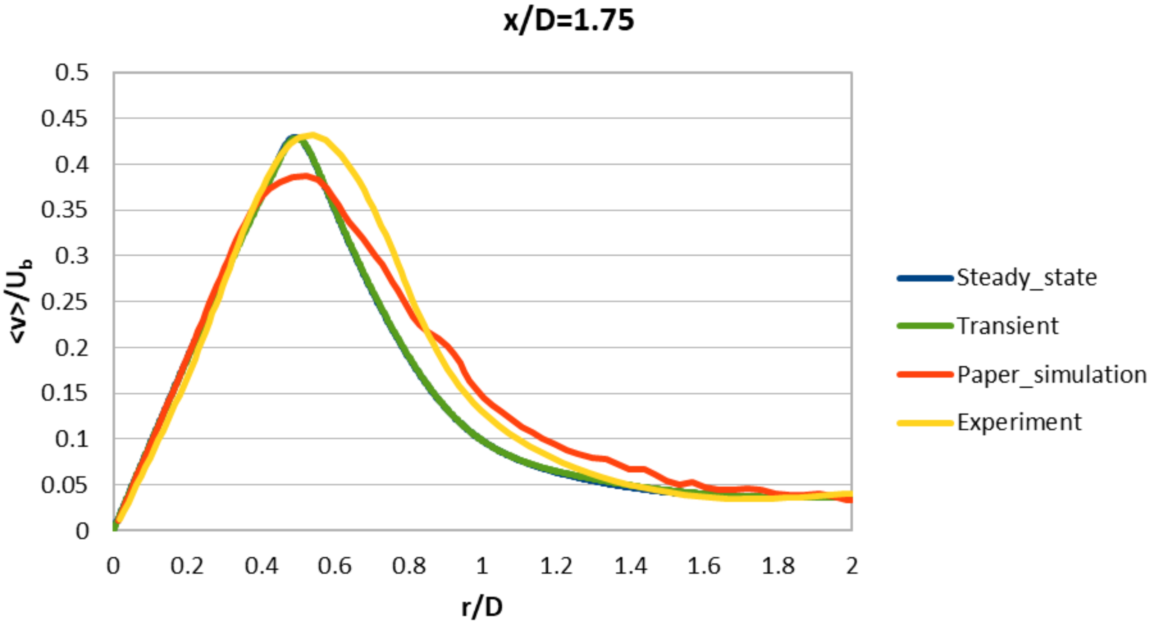


Figure 57. *Comparison of the mean radial velocity profile at x/D=1.75, between the experimental data, the numerical simulations of the authors and the thesis' numerical simulations.*

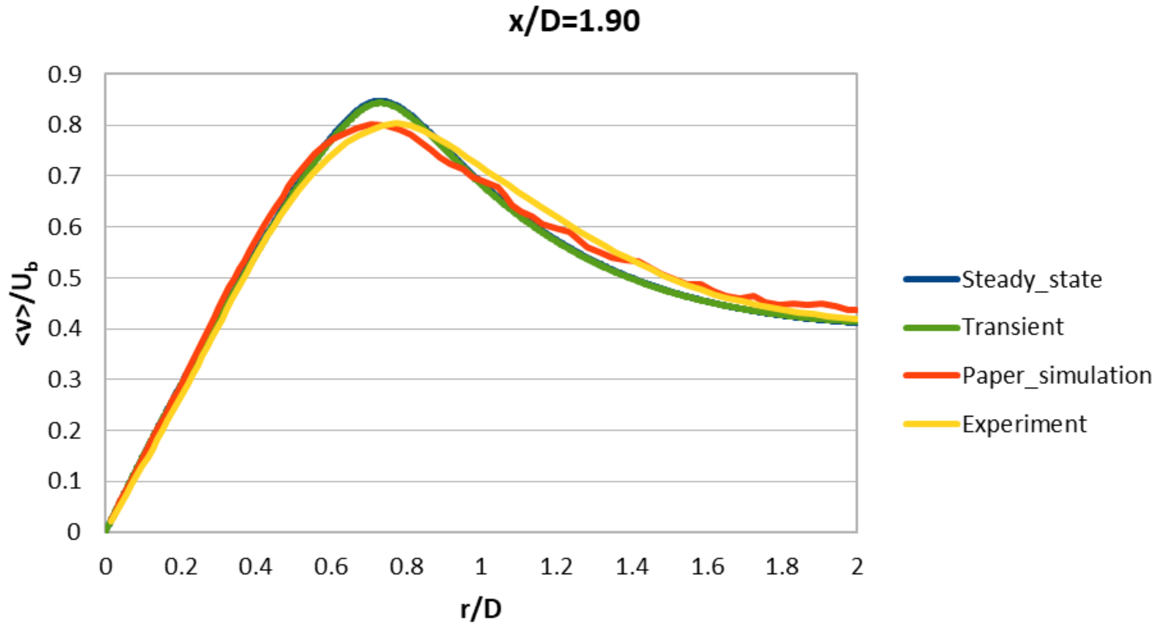


Figure 58. *Comparison of the mean radial velocity profile at x/D=1.90, between the experimental data, the numerical simulations of the authors and the thesis' numerical simulations.*

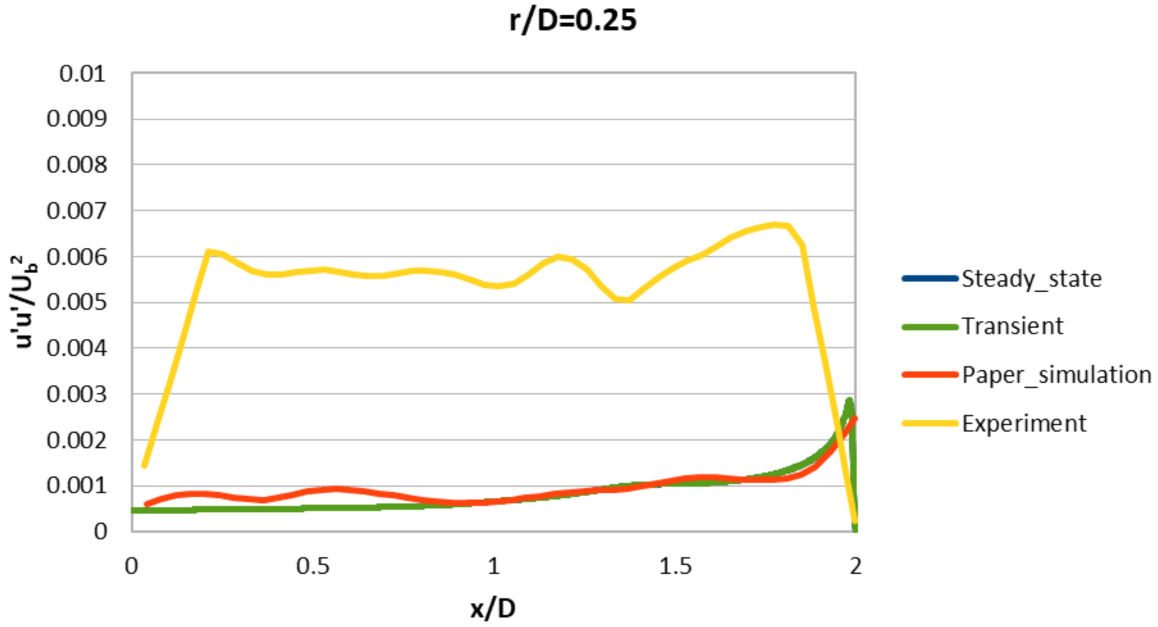


Figure 59. *Comparison of the Reynolds normal stresses at r/D=0.25, between the experimental data, the numerical simulations of the authors and the thesis' numerical simulations.*

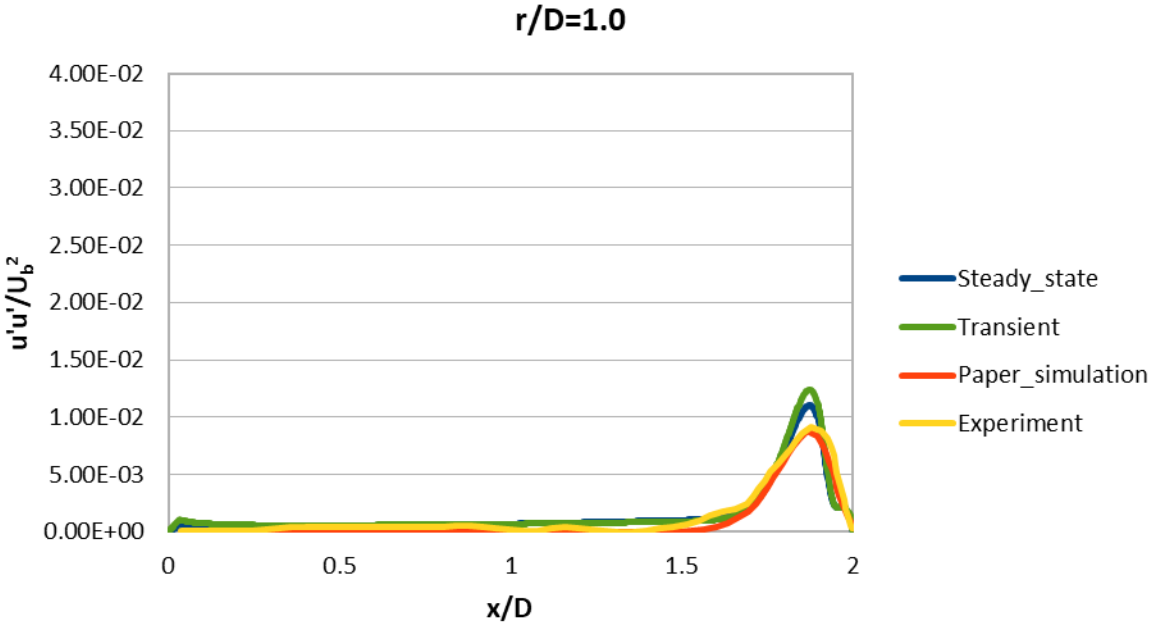


__Figure 60.__ Comparison of the Reynolds normal stresses at r/D=1.0, between the experimental data, the numerical simulations of the authors and the thesis' numerical simulations.

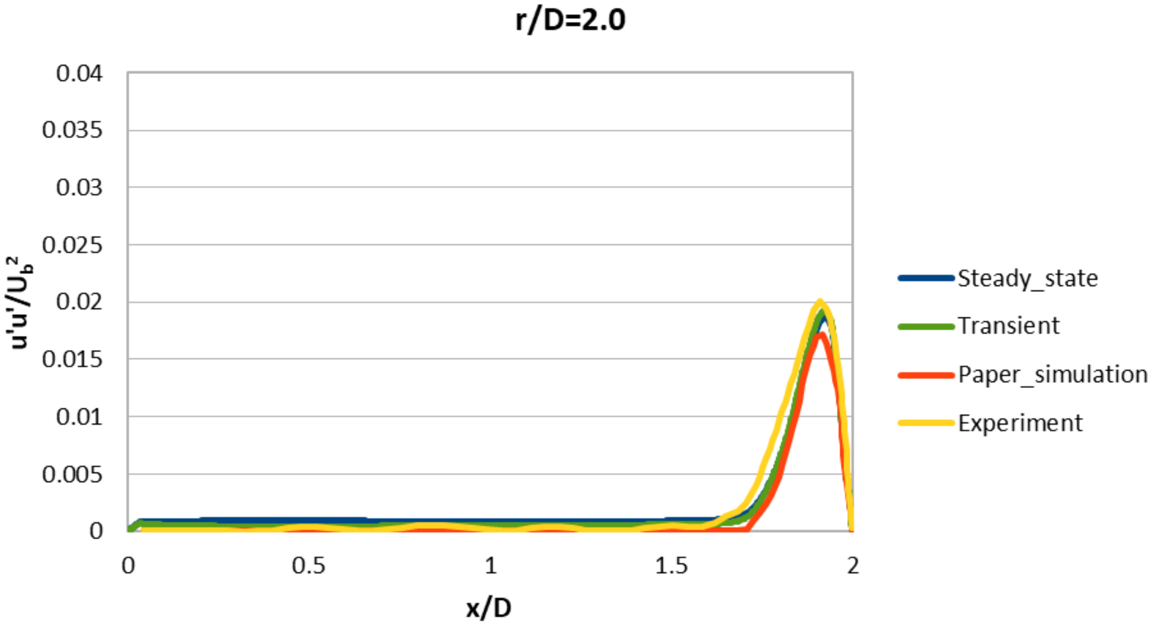


__Figure 61.__ Comparison of the Reynolds normal stresses at r/D=2.0, between the experimental data, the numerical simulations of the authors and the thesis' numerical simulations.

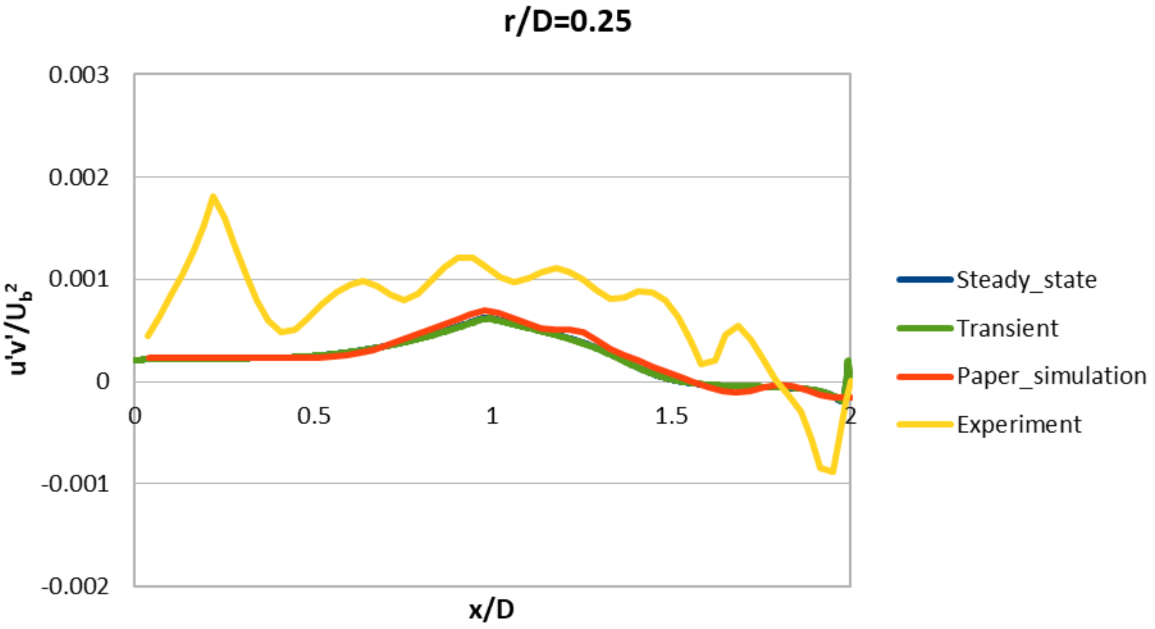


Figure 62. *Comparison of the Reynolds shear stresses at r/D=0.25, between the experimental data, the numerical simulations of the authors and the thesis' numerical simulations.*

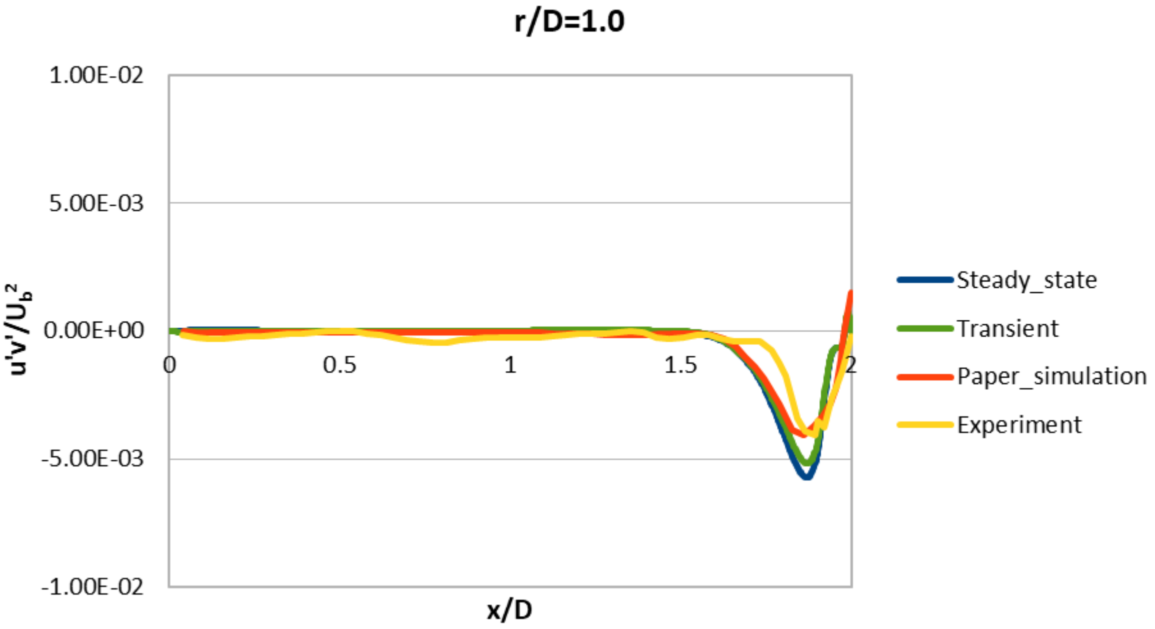


Figure 63. *Comparison of the Reynolds shear stresses at r/D=1.0, between the experimental data, the numerical simulations of the authors and the thesis' numerical simulations.*

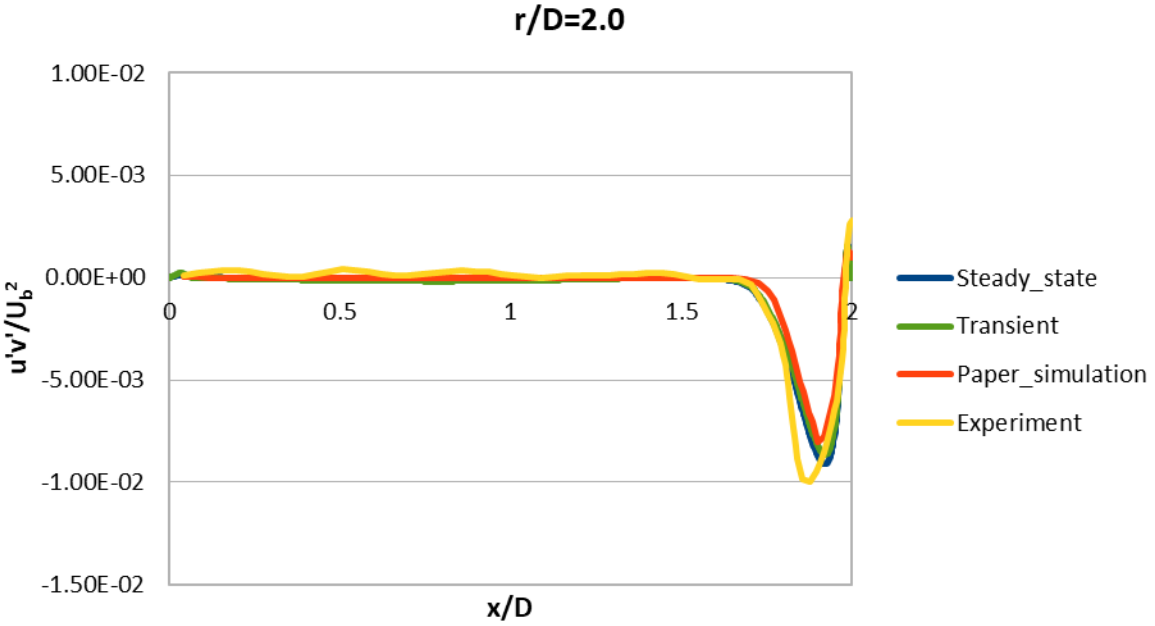


Figure 64. *Comparison of the Reynolds shear stresses at r/D=2.0, between the experimental data, the numerical simulations of the authors and the thesis' numerical simulations.*

3.8 Results overview

In general, this time results seemed to be both qualitatively and quantitatively accurate. Concerning the flow field of figure 50, its characteristics were as expected. The flow presented all the qualitative characteristics which were described at section 2.6, previously. In the left part of the flow field, a recirculation zone occurred above the wall-jet region, as expected from literature [12], [68]. In classic hydrodynamics, it is assumed to be created due to the separation of the boundary layer and the subsequent pressure gradients. The main issue of the previous 2D simulations, which was that the accelerating zone inside the wall-jet region was much bigger than it should be, seems to have been got over, at a first glance. This is also confirmed from the results of the variables, below in the diagrams. It must be pointed out that the paper's and the thesis' numerical simulations came in very good agreement both qualitatively and quantitatively, except for figure 56, in which major differences are observed in the accuracy. The same goes for the steady-state and the transient simulations carried out, the results of whom are almost identical for all diagrams. The Co_{max} values showed that the solution was totally stable throughout the time-steps of the transient simulation, with its maximum value equaling 0.850447<1. For the wall shear stresses, despite the good agreement with the experimental data, some fluctuations existing at the area of 1<r/D<2.5 could not be captured by the simulations. The mean axial velocities were adequately captured by the simulations, with the biggest deviation from the experiment met at the closest distance from the impinging target. In this case, the maximum deviation was met at approximately r/D=0.95, equaling $<u>/U_b=0.1$. Speaking of the radial velocities, as said previously, only the location of x/D=1.50 showed remarkable differences. There, apart from the accuracy, some difference in the location of the minima and maxima was detected. These locations were displaced left, relative to the experimental data. Moreover, at r/D=0.25, apart from some accuracy missing with respect to the experimental findings, the turbulent fluctuations in the Reynolds normal stresses could not be captured. However, unlike the simulations of the authors, our simulations could capture the sudden drop happening at x/D=2. At the same location, the models could also not detect the turbulent fluctuations in the Reynolds shear stresses.

3.9 Conclusions

All in all, for both the steady-state and the transient simulations, the validation and the verification of the results were achieved. To start with, the 3D simulations conducted, confirm the popular belief that 2D simulations are unable to offer accuracy in turbulent flows. Moreover, the similarity of the findings between the steady-state and the transient simulations led us to confirm the steady-state nature of the problem. The minor differences between the thesis' and the paper's simulation results, can be attributed to different factors. A contributing factor for sure is the differences in the algorithms used in each CFD software. For example, the authors of the publication used the pressure-based coupled algorithm and not a segregated algorithm like in our case. In addition, there were differences between the choice of the mesh, the algorithm control parameters and the discretization schemes. The last difference was in the boundary conditions used, as in the paper, authors used only a part of the domain, and thus, used symmetrical conditions in some boundaries. However, although the differences were major between the simulations, as mentioned before, these settings cannot change the results of a simulation dramatically. Also, the different types of errors that inevitably exist when using numerical methods can be blamed for the different findings between our results and the experimental ones. Specifically, the introduction of models for the description of phenomena, like turbulence, includes uncertainty that make it diverge from the exact solution. Sources of uncertainty can be [75]:

- Not thorough understanding of the physical phenomena
- Uncertainty in some parameters of the problem
- Not enough experimental confirmation of the problem
- Model simplifications

Discretization errors also contribute to inaccuracy, being maybe the most important factor. They are defined as the difference between the analytical solution of the equations and the numerical one, which comes from the discretization of the differential equations. The discretization process leads to an iterative procedure, which solves the discretized equations until the convergence criteria is reached. Nevertheless, even if the convergence point is reached, the exact solution of these equations is never reached, due to rounding errors caused by the machine precision of the computer. Such errors are called iteration errors [76]. These errors can also be responsible for the asymmetry in the results of the simulations, since at high-Reynolds number, flows become extremely sensitive to small changes [77]. Other errors that usually cause accuracy issues, are programming and user errors, that relate to code mistakes. Moreover, a finer mesh closer to the bottom face of the geometry with a respective y^+ value approximating 1, could probably offer greater accuracy to the results, according to literature. However, because of the already big computational domain and the restrictions on the computational power of the computer in which the simulations were run, that value could not be achieved. Concerning the inability of the numerical simulations to accurately compute the axial velocities at x/D=1.90, this is assumed to be due to the anisotropy of turbulence nearer to the impingement surface, where turbulence is increased. Actually, in many RANS models anisotropy effects are neglected, since most of them are based on equilibrium and isotropic turbulence. However, this is not the case in problems with strong adverse pressure gradients, curvatures and complex geometries [78]. Apart from that, the flow itself is anisotropic, since after jet impingement velocity becomes almost completely two-dimensional, with the axial component practically vanishing.

Although all the above factors can contribute to the accuracy of the solution, the role of the method a turbulent flow is solved, is maybe the most significant. When the RANS approach is used, the mean flow properties are solved and not the instantaneous, eliminating the turbulence-related physics from the equations, and demanding the introduction of turbulence models to account for them. Large and small-scale turbulence is not resolved, but is instead modelled via the Reynolds stress term. This way, turbulent fluctuations are not accurately computed, leading to an overall inaccuracy of the flow field. This is the main reason that the biggest deviations in the results, and particularly in the Reynolds stresses plots happen for r/D=0.25, which is the area closest to the impingement surface. There go the steepest gradients, that cannot be accurately captured by RANS models. Another reason for the deviations in this area is the fact that uniform turbulence intensity was assumed over the nozzle radius, while in reality, intensity peaks near the walls. However, other more accurate methods exist, such as the Direct Numerical Simulation (DNS) and the Large-Eddy Simulation (LES). In the former method, both large and small-scale eddies down to the Kolmogorov scale are fully resolved in time and space [79], leading to the need of extreme computational power for simple geometries and low-Reynolds numbers. In the latter method, only the largest scales are fully resolved, while the smallest ones are modeled, and although the computational costs are less than those in DNS, they are still considered expensive. In figure 65, the flow field of a turbulent jet using RANS, LES and DNS is illustrated. Engineers usually use RANS for turbulent CFD simulations, because their focus is on mean characteristics such as forces, velocities etc. For very complex geometries and high Reynolds numbers they sometimes use LES if RANS do not give accurate results, while DNS is only used for academic research, in order to get a better understanding of turbulence. As a result, although more accurate results could be obtained in the 3D simulations, for the specific case in which results are in good agreement with the experimental ones, it is not computationally efficient.

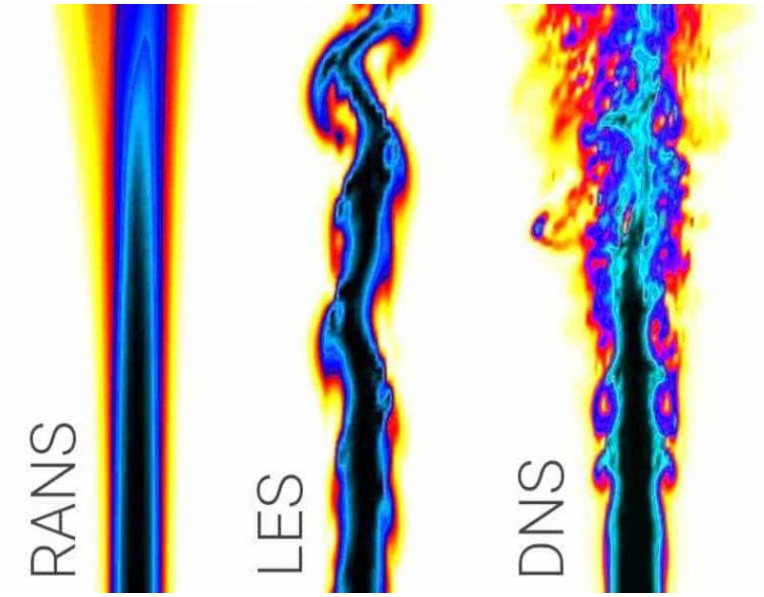


Figure 65. *CFD modeling of a turbulent jet using RANS, LES and DNS [32].*

Unit 4. **PARAMETRIC ANALYSIS OF 3D TURBULENT IMPINGING JETS**

4.1 The 3D parametric cases

After the validation and verification of the previous 3D simulations, a parametric analysis on the same 3D case was carried out. The goal of this analysis was to test whether or not the change on Reynolds numbers affected the development of the flow field near the impingement target. The simulations were run using the steady-state setup of the previous case, where four different bulk velocities and four different kinematic viscosities were addressed. The first four simulations involved the bulk velocity change. Bulk velocity varied between U_b=4.06 m/s, U_b=5.7 m/s, U_b=9.75 m/s and U_b=12.2 m/s, with the inlet velocity profiles finally turning into:

- $U_{in} = 5 \cdot (1 - r \ / \ R)^{1/5}$
- $U_{in} = 7 \cdot (1 - r \ / \ R)^{1/5}$
- $U_{in} = 12 \cdot (1 - r \ / \ R)^{1/5}$
- $U_{in} = 15 \cdot (1 - r \ / \ R)^{1/5}$

Apart from the inlet velocity boundary conditions change, the velocity changes demanded different values for k, omega and nut inlet boundary conditions. Expressions *(1.52)-(1.56)* were used for the determination of their values, with the change being introduced due to the bulk velocity involved in the definition of k. Concerning the four rest simulations involving the fluid's kinematic viscosity change, no additional parameters needed change. The different values studied were v=8e-7 m^2/s, v=1.639e-6 m^2/s, v=4e-6 m^2/s and v=1.5443e-5 m^2/s. These values were chosen because they approach the kinematic viscosities of liquids and dilute liquid-solid mixtures met in impinging jet applications. Often, multi-phase flows with the solid-phase in form of dust, are considered to be made up of a single-phase with composed properties. The first value of the kinematic viscosity studied corresponds to water in 30°C, the second to a typical slurry flow, the third to diesel fuel in 25°C and the last one to the steady-state simulation addressed in chapter 3. The resulting Reynolds numbers in the total eight simulations are gathered in table 9.

Table 9. *Resulting Reynolds number in the different flow regimes.*

U_b=4.06 m/s	Re=9,727	**v=8e-7 m^2/s**	Re=444,000
U_b=5.7 m/s	Re=13,656	**v=1.639e-6 m^2/s**	Re=216,717
U_b=9.75 m/s	Re=23,360	**v=4e-6 m^2/s**	Re=88,800
U_b=12.2 m/s	Re=29,230	**v=1.5443e-5 m^2/s**	Re=23,000

All the simulations involving bulk velocity change were run for 15,000 iterations, reaching convergence criteria analogous to the initial simulations of chapter 3. Concerning the simulations of the kinematic viscosities change, some of them were either run for 5,000 using first order discretization schemes and 10,000 iterations with second order schemes, or used the results of the steady-state simulation of chapter 3 as initial conditions and were run for 5,000 iterations in total. Both modifications helped in increasing solution stability, as initially (running for 15,000 iterations with second order discretization schemes) there were divergence issues. The results that follow below focus on the regions closest to the impingement location, i.e., x/D=1.90 and r/D=0.25. The wall-shear stress is illustrated in figure 66, the axial and radial velocities in figures 67 and 68, respectively, the Reynolds normal stresses in figure 69 and the Reynolds shear stresses in figure 70. In general, apart from the changes in some boundary conditions as described above, the setup of the simulations remained untouched.

4.2 Parametric analysis results

For the results depicted below, it must be clarified that as U_b, the bulk velocity applied in each case was attributed.

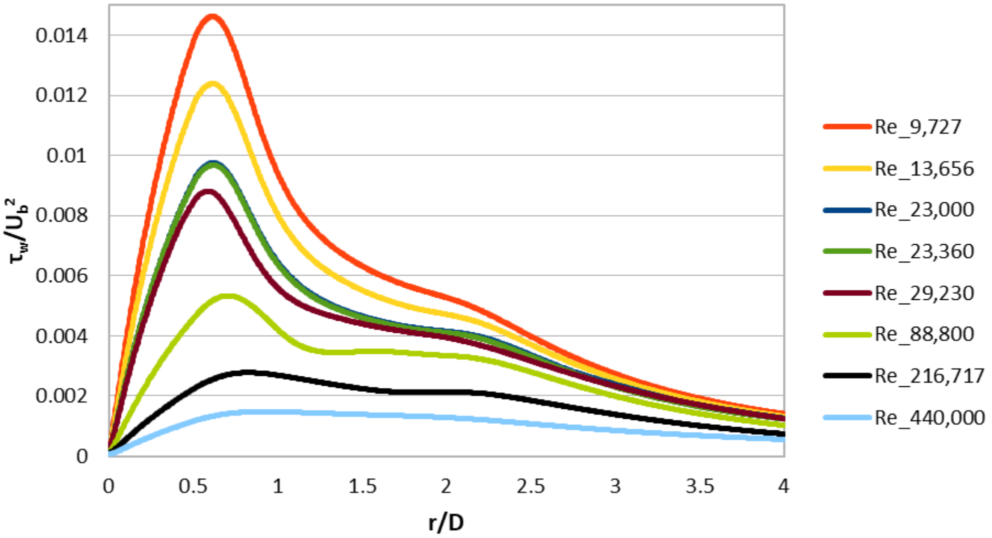


Figure 66. *Comparison of the wall shear stress profile between the different Reynolds numbers.*

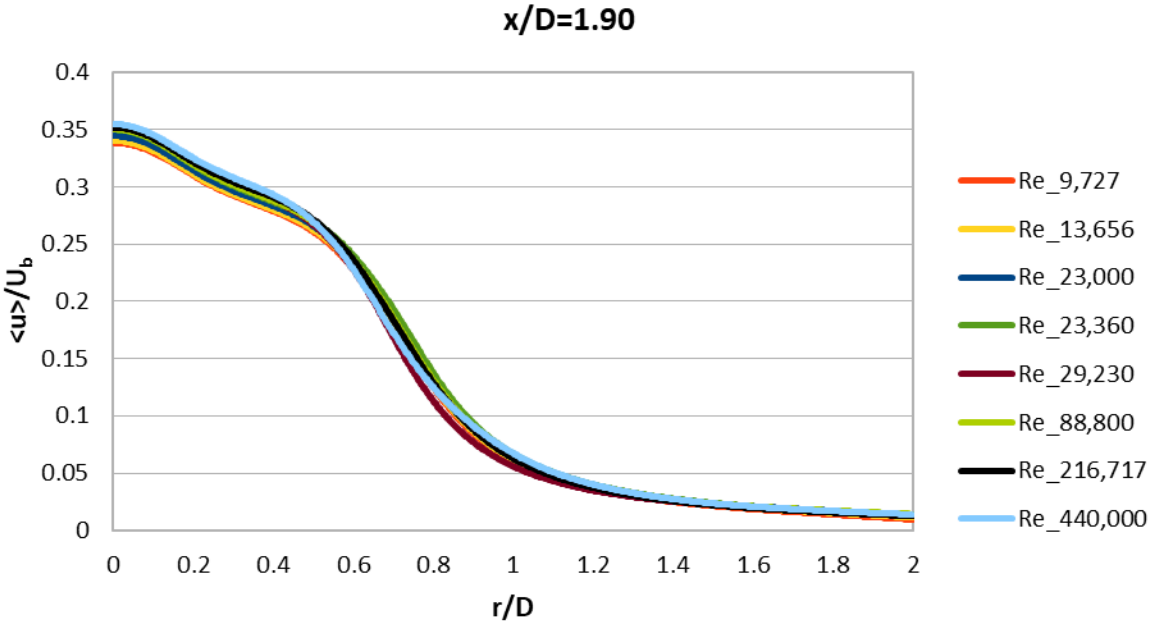


Figure 67. *Comparison of the mean axial velocity profile at x/D=1.90, between the different Reynolds numbers.*

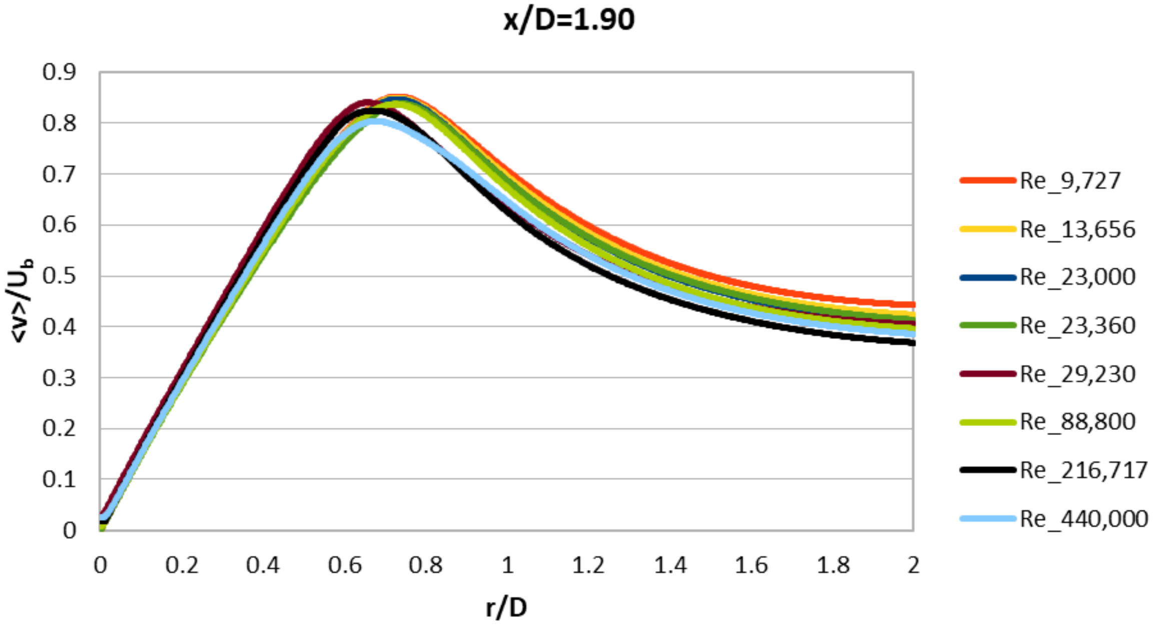


Figure 68. *Comparison of the mean radial velocity profile at x/D=1.90, between the different Reynolds numbers.*

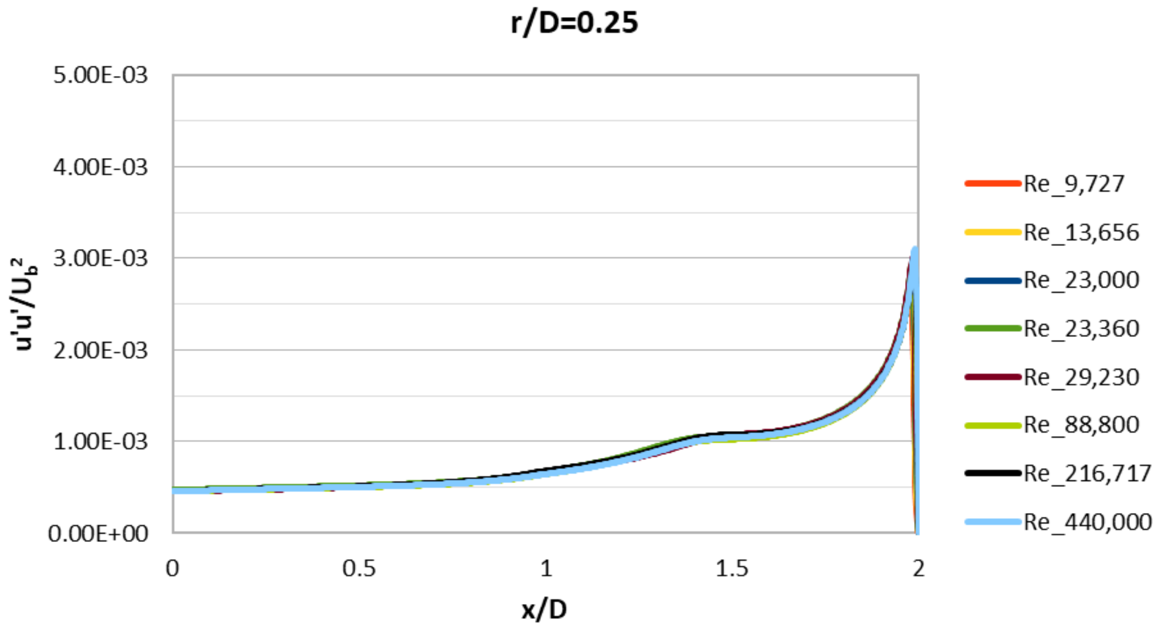


Figure 69. *Comparison of the Reynolds normal stresses at r/D=0.25, between the different Reynolds numbers.*

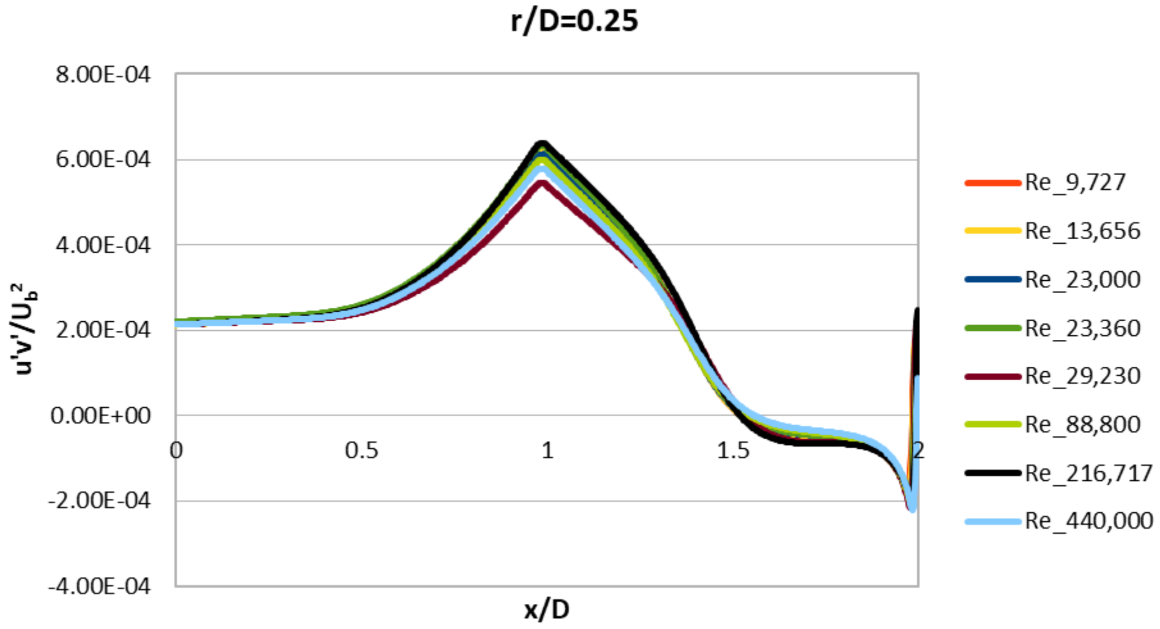


Figure 70. *Comparison of the Reynolds shear stresses at r/D=0.25, between the different Reynolds numbers.*

4.3 Results overview and conclusions

At a first glance, it is obvious that wall-shear stress was the only variable of those studied above that greatly depends on Re. The fact that the rest of the plots almost coincided or had very minor differences between them, shows that the respective profiles were almost independent of Re, and it was only their magnitude that changed. By looking at figure 66, it can be seen that by increasing Re, wall-shear stress profile became much smoother. Special reference should be given to figure 68. There, the two smallest kinematic viscosities together with the largest bulk velocity showed a slight left displacement of the plots' maxima. However, Re=88,800 which is larger than Re=29,230 (largest bulk velocity) did not illustrate the same displacement, and as a result, no clear conclusion can be drawn for the impact of Re. The above results can be physically interpreted. According to literature, Re captures the ratio of the effects of inertial forces to viscous forces. When the flow is fully turbulent, the effects of viscosity are low and the flow becomes practically independent of Re. This is especially the case, the further we go with respect to solid boundaries. In solid boundaries there is always a boundary layer where viscosity cannot be neglected, since viscous effects dominate. This is the reason that wall-shear stress seems to be dependent on Re, since it's a property calculated in solid surfaces.

Unit 5. **GENERAL CONCLUSIONS AND POTENTIAL FUTURE WORK**

5.1 General conclusions

Some cases involving single-phase turbulent impinging jets have been tested, with the aim of investigating their flow characteristics, which need more insight. The validation of the first experimental case has not been successful, due to the fact that the simulations were carried out in two dimensions. Although some of the results have shown both qualitative and quantitative results, the majority of them lack the latter, as expected from literature. The case has been found to be mesh-independent, assuring once again that the whole y^+ range (0-300) can be applied to capture the physics of the problem, when using the k-ω SST turbulence model with the appropriate wall-functions, in the case its high-Reynolds form is applied. On the contrary, both the validation and verification of the following 3D case has been achieved. Our simulation results have come in very good agreement with the reference paper's results, with a flaw existing in all the numerical simulations, being the inability to capture some fluctuations at the wall-shear stress and Reynolds stresses plots. This is assumed to be mainly due to the RANS approach adopted, which lacks in accuracy. Concerning the transient simulation, the good results obtained show that it can be used in a respective multi-phase turbulent impinging jet study as described at section 5.2, below. Last but not least, the parametric analysis has shown that the wall-shear stress has been the only variable which depends on Re, because of the fact that in solid boundaries viscous effects dominate. Velocity and Reynolds stress profiles have been found to be Re independent, with their magnitude being the only one which changes, through the change on either bulk velocity or kinematic viscosity of the fluid. This is assumed to be due to the fully turbulent flow, which makes them practically independent of Re.

5.2 Future work

The outcomes of this thesis can subsequently be used as a tool for another research, this time concerning multi-phase turbulent impinging jets. Specifically, they can be the foundation for the study of these jets and their use as a controlled erosion environment, where material properties can be studied and models can be developed for the prediction of impact erosion. Erosion is a major problem that concerns many industries, i.e., the mining and the oil and gas industry. In such industries, it is caused by the impingement of particle and hydrocarbon fluid mixtures onto material surfaces during production, transportation and processing operations [80]. Together with the cavitation of liquid-droplets, they make up the main sources of equipment failure. By saying «equipment», it is meant components, such as pipelines, choke valves etc. The price of such failures is time-consumption and high maintenance costs; therefore, their avoidance is crucial. In addition, as mentioned in chapter 1, such flows involve in slurry flows and wind turbines. As a result, the former can by its own constitute a study in a future thesis, in which erosion can be investigated after the discharge from the pump to the surrounding structure.

Speaking of erosion, a broad literature review has been carried out the last century. Specifically, Meng and Ludema [81] reviewed over 5000 papers from 1957 to 1992, in which they identified 28 different erosion models among the almost 2000 existing empirical models, a sign of the poor agreement between authors [82]. The big number of different forms of the equations is said to exist, due to the fact that erosion is a phenomenon that is affected by over 20 different factors. Until today, the factors known to influence erosion are divided into three categories that relate to the particles, the surfaces and the carrier fluid and are gathered in table 10:

Table 10. *Factors affecting erosion [82] .*

For particles	For surfaces	For the carrier fluid
Impact and rebound angles	Physical properties	State of motion (laminar or turbulent)
Impact and rebound speeds	Change in shape caused by erosion	Velocity
Rotation before and after impact	Stress level	Temperature
Shape and size	Temperature	Chemical composition and physical properties
Volume concentration	Presence of oxide coatings	
Physical properties (hardness strength and density)	Simultaneous occurrence of corrosion	
Fragmentation		
Interactions (with surfaces, fluid or other particles)		
Temperature		
Presence of additives		
Electrical charge		

It is worth mentioning that authors constantly add new factors in their theoretical approach and this is why many of the existing models are presented as a «personal opinion».

By the above sayings, one draws the conclusion that more insight on the topic must be provided, in order to obtain a better understanding of multi-phase flows and the way the different parameters and properties of both the fluid and the particles affect wear. To contribute to that, a future thesis can involve analogous geometries like those studied in the current thesis, where together with the liquid, particles will be inserted into the pipe. For the numerical simulations, *OpenFOAM, Fluent* or another software that can deal with multi-phase flows can be used to solve the flow and compute erosion. Specifically, DEM (Discrete Element Method) coupled with CFD (CFDEM) is a powerful technique used to study, design and optimize multi-phase flows [83]. *OpenFOAM* can take over the fluid dynamics part of the flow, while the open-source DEM particle simulation software *LIGGGHTS* (LAMMPS improved for general granular and granular heat transfer simulations) [84] can take over the solid phase. In this case, the two phases are separately solved and information pass from one software to the other. Solids may be handled in a Lagrangian way, defining the forces acting on the discrete phase and obtaining velocities and positions by means of integration of Newton’s equations. The Lagrangian approach can be subdivided with respect to the kind of coupling between the particles and the fluid, into:

- One-way coupling, if there is a low concentration of particles or the particle diameter is very small. In this case the impact of particles on the fluid is neglected.
- Two-way coupling, if the impact of particles on the fluid is no longer neglected, so a new term should be included in the equations.
- Four-way coupling, where particle-particle interactions are also taken into account. This case is often applied in large particle diameters and large particle concentrations.

Also, two more different approaches have been adopted for CFDEM coupling: resolved and unresolved coupling. Resolved coupling is applied when the particle size is bigger than the computational grid used for the CFD part, while unresolved coupling is applied in the opposite case. In both cases different algorithm implementations are used [85]. In addition, due to the nature of the flow, in CFDEM a transient simulation must be considered for the CFD part. This is the main reason why the transient simulation was carried out in this thesis. The good results obtained in chapter 3 show that it can make up the CFD part of a CFDEM simulation, in which particles will be inserted through the inlet. An unresolved coupling must be used for this case, due to the fine grid already used. An investigation on the effect that different parameters have on erosion can be done. Potential parameters studied can be the particle diameter, the particle concentrations, the particle shape, and the different particle and target materials. From then on, depending on the different diameters and concentrations involved, different kind of coupling must be applied to give accurate results and not neglect important forces, such as particle-particle contact forces, particle-wall forces, drag forces, buoyant forces etc. In *OpenFOAM* different solvers are available for CFDEM coupling, but in our case «cfdemSolverPiso» can be used for the unresolved coupling [86]. The results found will include particle-wall contact forces, particle velocities and positions, and after that, erosion equations existing in literature can be exploited to obtain impact erosion at the bottom. All in all, this is a potential case that can be studied, taking into consideration the findings of the current thesis.

REFERENCES

[1] D. T. New and S. C. Yu, Vortex Rings and Jets: Recent Developments in Near-Field Dynamics, Springer, 2015.

[2] S. Ashforth-Frost, K. Jambunathan and C. F. Whitney, "Velocity and Turbulence Characteristics of a Semiconfined Orthogonally Impinging Slot Jet," *Experimental Thermal and Fluid Science ,* no. 14, pp. 60-67, 1997.

[3] W. Grassi, "Thermopedia," 2011. [Online]. Available: https://thermopedia.com/jp/content/872/?tid=110&sn=. [Accessed 19 August 2022].

[4] N. Ilinca and B. Florin, "Impinging jets - a short review on strategies for heat transfer enhancement," *EENVIRO 2017 workshop - Advances in Heatand Transfer in Build Environment,* vol. 32, p. DOI: 10.1051/e3sconf/20183201013, 2018.

[5] "Anuj J., Shukla; Anupam, Dewan," *INTERNATIONAL JOURNAL OF HEAT AND TECHNOLOGY,* vol. 35, no. 1, pp. 153-166, 2017.

[6] D. McKendrick, S. R. Biggs, M. Fairweather and D. Rhodes, "Physical Modeling of Axisymmetric Turbulent Impinging Jets as used within the Nuclear Industry for Mobilization of Sludges," in *IYNC*, Interlaken, 2008.

[7] B. HAN and R. J. Goldstein, "Jet - Impingement Heat Transfer in Gas Turbine Systems," *The New York Academy of Sciences,* vol. 934, no. 1, pp. 147-161, 2006.

[8] K. Cameron, "CYCLE WORLD," 12 December 2018. [Online]. Available: https://www.cycleworld.com/how-do-piston-cooling-oil-jets-work/. [Accessed 19 August 2022].

[9] B. Kwon, T. Foulkes, T. Yang, N. Miljkovic and W. P. King, "Air Jet Impingement Cooling of Electronic Devices Using Additively Manufactured Nozzles," *IEEE Xplore,* vol. 10, no. 2, pp. 220-229, 2020.

[10] R. Viskanta, "Heat Transfer to Impinging Isothermal Gas and Flame Jets," *Experimental Thermal and Fluid Science,* vol. 6, no. 2, pp. 111-134, 1993.

[11] A. Sarkar, N. Nitin, M. V. Karwe and R. P. Singh, "Fluid Flow and Heat Transfer in Air Jet Impingement in Food Processing," *Journal of Food Science,* vol. 69, no. 4, pp. 113-122, 2006.

[12] K. J. Hammad and I. M. Milanovic, "Flow Structure in the Near - Wall Region of a Submerged Impinging Jet," *Fluids Engineering Division Summer Meeting,* vol. 1, pp. 1445-1455, 2009.

[13] M. Holger, "Heat and Mass Transfer between Impinging Gas Jets and Solid Surfaces," *Advances in Heat Transfer,* vol. 13, pp. 1-60, 1977.

[14] S.-M. Simionescu, N.-O. Tanase, D. Broboana and C. Balan, "Impinging Air Jets on Flat Surfaces at Low Reynolds Number," *Science Direct,* no. 112, pp. 194-203, 2017.

[15] "ANSYS," [Online]. Available: https://www.ansys.com/products/fluids/ansys-fluent. [Accessed 14 January 2023].

[16] A. Kelsey, B. Richard and T. Graham, "Modelling of Angled Jet Impingement on Large Flat Surfaces," in *Hazards XVIII: Process Safety - Sharing Best Practice*, Warrington, IChemE, 2004, pp. 419-435.

[17] "ANSYS," [Online]. Available: https://www.ansys.com/products/fluids/ansys-cfx. [Accessed 14 January 2023].

[18] L. M. Milne - Thomson, THEORETICAL AERODYNAMICS, New York: Dover Publications, 1973.

[19] P. Rubbert, "The impact of supercomputers on CFD," *Computing Systems in Engineering,* pp. 1-9, 1990.

[20] J. H. Ferziger and M. Peric, "Introduction to numerical methods," in *Computational Methods for Fluid Dynamics*, Springer, 2002, pp. 33-50.

[21] "Computer Hope," 31 December 2022. [Online]. Available: https://www.computerhope.com/jargon/b/byte.htm. [Accessed 06 February 2023].

[22] A. Liakopoulos, "Fundamental principles of fluid mechanics," in *Fluid Mechanics*, Volos, Tziola, 2019, pp. 27-37.

[23] T. Holzmann, "Holzmann CFD," 3 November 2019. [Online]. Available: https://holzmann-cfd.com/community/publications/mathematics-numerics-derivations-and-openfoam-free. [Accessed 25 August 2022].

[24] "COMSOL," 15 January 2015. [Online]. Available: https://www.comsol.com/multiphysics/navier-stokes-equations. [Accessed 26 August 2022].

[25] Shabbusharma, "PHYSICSWAVE," 22 March 2021. [Online]. Available: https://physicswave.com/gauss-divergence-theorem/. [Accessed 26 August 2022].

[26] "SIMSCALE," 03 September 2021. [Online]. Available: https://www.simscale.com/docs/simwiki/numerics-background/what-are-the-navier-stokes-equations/. [Accessed 06 September 2022].

[27] H. K. Versteeg and W. Malalasekera, "Forms of the general transport equations," in *An introduction to COMPUTATIONAL FLUID DYNAMICS - The finite volume method*, New Jersey, Pearson Education (US), 2007, pp. 25-34.

[28] T. Barth and M. Ohlberger, "Finite Volume Methods: Foundation and Analysis," *Encyclopedia of Computational Mechanics,* p. DOI:10.1002/0470091355.ECM010, 2004.

[29] S. P. Pope, "Introduction," in *Turbulent Flows*, Cambridge, 2000, pp. 3-9.

[30] P. M. Gerhart, A. L. Gerhart and J. I. Hochstein, "General Characteristics of Pipe Flow," in *Fluid Mechanics*, Wiley, 2018, pp. 383-389.

[31] F. U. Rahman, "THE CONSTRUCTOR-Building ideas," [Online]. Available: https://theconstructor.org/fluid-mechanics/laminar-turbulent-flow/559432/. [Accessed 22 October 2022].

[32] "ideal simulations," [Online]. Available: https://www.idealsimulations.com/resources/turbulence-models-in-cfd/. [Accessed 25 October 2022].

[33] "cadence- SYSTEM ANALYSIS," [Online]. Available: https://resources.system-analysis.cadence.com/blog/msa2022-examining-reynolds-number-for-turbulent-flow. [Accessed 22 October 2022].

[34] S. P. Pope, "An introduction to modelling and simulation," in *Turbulent Flows*, Cambridge, 2000, pp. 335-343.

[35] "cadence-SYSTEM ANALYSIS," [Online]. Available: https://resources.system-analysis.cadence.com/blog/msa2021-the-reynolds-averaged-navier-stokes-rans-equations-and-models. [Accessed 22 October 2022].

[36] Dey, Subhasish and R. Das, "Gravel-Bed Hydrodynamics: Double Averaging Approach," *Journal of Hydraulic Engineering,* vol. 138, no. 8, pp. 25-707, 2012.

[37] "American Meteorological Society," 20 February 2012. [Online]. Available: https://glossary.ametsoc.org/wiki/Reynolds_stresses. [Accessed 23 October 2022].

[38] N. A. Che Sidik, S. N. Akmal Yusof, Y. Asako, M. S.B and A. Aziz, "A Short Review on RANS Turbulence Models," *CFD Letters,* vol. 12, no. 11, pp. 83-96, 2020.

[39] "All About CFD," 4 October 2021. [Online]. Available: https://cfdisrael.blog/2021/10/04/eddy-viscosity-concept-turbulent-kinetic-energy-equation/. [Accessed 24 October 2022].

[40] F. Menter, "Two-equation eddy-viscosity turbulence models for engineering applications," *AIAA Journal,* vol. 32, no. 8, pp. 1598-1605, 1994 .

[41] "Wikipedia," 10 May 2022. [Online]. Available: https://en.wikipedia.org/wiki/Law_of_the_wall. [Accessed 9 December 2022].

[42] W. Frei, "COMSOL," 6 July 2017. [Online]. Available: https://www.comsol.com/blogs/which-turbulence-model-should-choose-cfd-application/. [Accessed 5 December 2022].

[43] "CFD online," 28 February 2011. [Online]. Available: https://www.cfd-online.com/Wiki/SST_k-omega_model. [Accessed 5 December 2022].

[44] "CFD online," 15 January 2014. [Online]. Available: https://www.cfd-online.com/Wiki/Turbulence_free-stream_boundary_conditions. [Accessed 08 February 2023].

[45] [Online]. Available: https://openfoam.org/. [Accessed 11 December 2022].

[46] "cplusplus," [Online]. Available: https://cplusplus.com/. [Accessed 27 January 2023].

[47] "Wikipedia," 24 September 2022. [Online]. Available: https://en.wikipedia.org/wiki/OpenFOAM. [Accessed 10 December 2022].

[48] K. G. Mooney, T. Maric and J. Hopken, The OpenFOAM Technology Primer, Dublin: Sourceflux, 2014.

[49] C. Greenshields, "CFD Direct," 12 July 2022. [Online]. Available: https://doc.cfd.direct/openfoam/user-guide-v10/introduction. [Accessed 22 December 2022].

[50] C. DeGroot, "MAPLE KEY LABS," 25 August 2020. [Online]. Available: https://maplekeylabs.com/understanding-the-openfoam-case-structure/. [Accessed 11 December 2022].

[51] "OpenFOAM," [Online]. Available: https://www.openfoam.com/documentation/user-guide/2-openfoam-cases/2.1-file-structure-of-openfoam-cases. [Accessed 12 December 2022].

[52] M. Atkins, "Velocity Field Measurement Using Particle Image Velocimetry (PIV)," in *Application of Thermo-Fluidic Measurement Techniques*, Butterworth-Heinemann, 2016, pp. 125-166.

[53] "ParaView," [Online]. Available: https://www.paraview.org/. [Accessed 14 December 2022].

[54] "Ubuntu," [Online]. Available: https://ubuntu.com/. [Accessed 14 December 2022].

[55] "SIMSCALE," 12 September 2022. [Online]. Available: https://www.simscale.com/docs/simulation-setup/meshing/mesh-quality/. [Accessed 23 January 2023].

[56] F. Menter, M. Kuntz and R. Langtry, "Ten years of industrial experience with the SST turbulence model," in *Heat and Mass Transfer*, Springer, 2003, pp. 1-8.

[57] "Openfoamwiki," 16 September 2022. [Online]. Available: https://openfoamwiki.net/index.php/SimpleFoam. [Accessed 14 December 2022].

[58] "Simscale," 25 May 2021. [Online]. Available: https://www.simscale.com/docs/simulation-setup/global-settings/k-omega-sst/. [Accessed 15 December 2022].

[59] "OpenFOAM: User Guide v2112," [Online]. Available: https://www.openfoam.com/documentation/guides/latest/doc/guide-bcs-wall-turbulence-kqRWallFunction.html. [Accessed 15 December 2022].

[60] "OenFOAM: User Guide v2112," [Online]. Available: https://www.openfoam.com/documentation/guides/latest/doc/guide-bcs-wall-turbulence-omegaWallFunction.html. [Accessed 15 December 2022].

[61] C. Greenshields, "CFD direct," 10 July 2018. [Online]. Available: https://doc.cfd.direct/openfoam/user-guide-v6/fvschemes. [Accessed 15 December 2022].

[62] "Manchester CFD," [Online]. Available: https://www.manchestercfd.co.uk/post/some-very-important-terms-any-cfd-user-should-know. [Accessed 15 December 2022].

[63] "OpenFOAM," [Online]. Available: https://www.openfoam.com/documentation/user-guide/6-solving/6.3-solution-and-algorithm-control. [Accessed 17 December 2022].

[64] "LibreOffice," [Online]. Available: https://www.libreoffice.org/. [Accessed 23 January 2023].

[65] "Engauge Digitizer," [Online]. Available: https://markummitchell.github.io/engauge-digitizer/. [Accessed 29 Dcember 2022].

[66] N. E. T. Laboratory, "2D versus 3D CFD Simulations of Circulating Fluidized Bed Risers," in *Multiphase Flow Workshop*, Morgantown, 2012.

[67] "Wikipedia," 22 April 2021. [Online]. Available: https://en.wikipedia.org/wiki/Vortex_stretching. [Accessed 31 December 2022].

[68] A. U. Zahir, A. A. M. Yasir and M. T. Miccal, "The effect of inflow conditions on the development of non-swirling versus swirling impinging turbulent jets," *Computers & Fluids,* vol. 118, pp. 255-273, 2015.

[69] "OpenFOAM," [Online]. Available: https://www.openfoam.com/documentation/user-guide/a-reference/a.4-standard-boundary-conditions. [Accessed 03 January 2023].

[70] "OpenFOAM: User Guide v2112," 2016. [Online]. Available: https://www.openfoam.com/documentation/guides/latest/doc/guide-schemes-time-crank-nicolson.html. [Accessed 5 January 2023].

[71] "Wikipedia," [Online]. Available: https://en.wikipedia.org/wiki/PISO_algorithm. [Accessed 05 January 2023].

[72] "Wolf Dynamics," [Online]. Available: http://www.wolfdynamics.com/wiki/tipsandtricks.pdf. [Accessed 05 January 2023].

[73] G. Camihna, "SIMSCALE," 17 October 2019. [Online]. Available: https://www.simscale.com/blog/cfl-condition/. [Accessed 6 January 2023].

[74] "Gnuplot," [Online]. Available: http://www.gnuplot.info/. [Accessed 11 January 2023].

[75] NASA, "NPARC Alliance CFD Verification and Validation," 10 February 2021. [Online]. Available: https://www.grc.nasa.gov/www/wind/valid/tutorial/errors.html. [Accessed 10 January 2023].

[76] J. Ferziger and M. Peric, "Efficiency and Accuracy Improvement," in *Computational Methods for Fluid Dynamics*, Springer, 2002, pp. 368-409.

[77] S. B. Pope, "The statistical description of turbulent flows," in *Turbulent Flows*, Cambridge University, 2000, pp. 34-82.

[78] Y. Hao, L. Yangwei and L. Lipeng, "Turbulence anisotropy analysis in a highly loaded linear compressor cascade," *Aerospace Science and Technology,* vol. 91, pp. 241-254, 2019.

[79] J. Ferziger and M. Peric, "Turbulent Flows," in *Computational Methods for Fluid Dynamics*, Springer, 2002, pp. 302-346.

[80] J. T. Tortor, "MATERIAL SOLUTIONS TO REDUCE WEAR AND EROSION IN OIL AND GAS INDUSTRY," *International Journal of Applied Science,* vol. 5, p. 61, 2018.

[81] H. Meng and K. Ludema, "Wear models and predictive equations: their form and content," Vols. 181-183, pp. 443-457, 1995.

[82] A. Lopez, M. T. Stickland and W. M. Dempster, "CFD study of fluid flow changes with erosion," *Computer Physics Communications,* vol. 227, pp. 27-41, 2018.

[83] A. Dr. Hager, C. Dr Kloss and C. Dr. Goniva, "Combining Open Source and Easy Access in the field of DEM and coupled CFD-DEM: LIGGGHTS®, CFDEM®coupling and CFDEM®workbench," *Computer Aided Chemical Engineering,* vol. 43, pp. 1699-1704, 2018.

[84] "CFDEM PROJECT," [Online]. Available: https://www.cfdem.com/liggghts-open-source-discrete-element-method-particle-simulation-code. [Accessed 12 January 2023].

[85] D. Fantin, "TUDelft," 31 October 2018. [Online]. Available: https://repository.tudelft.nl/islandora/object/uuid:6ddce0bf-dc00-4f7a-bb07-040b9851a8e8?collection=education. [Accessed 23 January 2023].

[86] "CFDEMcoupling v3.X," [Online]. Available: https://www.cfdem.com/media/CFDEM/docu/cfdemSolverPiso.html. [Accessed 14 January 2023].