National Technical University of Athens
School of Naval Architecture and Marine Engineering
Laboratory of Ship and Marine Hydrodynamics

# Hydrodynamic optimization using CFD methodologies with open-source software

Diploma thesis

**Kourtis Anastasios**

Thesis Supervisor: G. Papadakis, Assistant Professor

Comittee Member: G. Zaraphonitis, Professor

Comittee Member: G. Grigoropoulos, Professor

July 2023, Athens

# Contents

# List of Figures

# List of Tables

# Acknowledgements

I would like to express my sincerest gratitude to all those who have supported and inspired me during the course of my research and writing. First and foremost, I would like to express my deepest appreciation to my thesis advisor Professor G. Papadakis, for his invaluable guidance, support, encouragement and patience throughout this journey. His expertise and mentorship have been instrumental in shaping my research and helping me bring my work to its current form. I would like to acknowledge the contributions of all those who have directly or indirectly influenced my research, including my colleagues, fellow researchers, and participants in my study.

Finally, I am grateful to my family and friends for their unwavering support and encouragement during the many ups and downs of this process. Their love and encouragement have been a source of strength and motivation throughout this journey, and without them, I would not be able to chase my dreams. This thesis would not have been possible without the support and encouragement of these individuals, and for this, I am deeply grateful.

# Abstract

The envision of the IMO for the decarbonization of the maritime industry has shifted the research efforts into more efficient designs and innovative solutions. This diploma thesis, facilitates this envision by examining the coupling of the open-source softwares OpenFOAM and OpenMDAO, for the application of trim optimization, in regards to minimizing the calm water resistance. Optimizing the trim can have significant implications for various aspects of vessel performance, including fuel efficiency. Also, it prepares the ground for multidisciplinary design optimization by demonstrating the feasibility of the method and subsequently developing the basic framework. The open access KCS bare hull geometry was employed, for which the Reynolds Averaged Navier-Stokes equations were solved to acquire the calm water resistance, in the OpenFOAM environment. For the geometry handling, a custom script was developed in Python language, and the optimization model was implemented in OpenMDAO, which also managed the data passage between the softwares and the script. Finally, the analysis was extended by updating the existing framework to perform a trivial shape form optimization at the bulbous area. Overall, the results showed a decrease in the total resistance, as expected, validating the methodology.

# Chapter 1

# Introduction

## 1.1 Introduction

The shipping industry has seen several developments throughout the last century the majority of which aimed at improving ships' safety and performance. Being responsible for the transportation of approximately 90% of consumer goods, it becomes apparent that the operating costs are enormous, of which the largest part is associated with the fuel needed for completing all those journeys across the globe. As observed from the events that occurred in the previous years, fossil fuels are prone to extreme fluctuations in prices, resulting in extra operating costs. Also, their applicability is questionable, as the industry is heading towards decarbonization. The fuel used for the operation of marine engines emits significant amounts of CO2 and other Greenhouse Gases (GHG) such as methane and nitrous oxide, which corresponds to 1,076 million tonnes in 2018 and accounts for 2.89% of global anthropogenic emissions, as presented in the Fourth IMO GHC Sutdy 2020 [34]. Although the shipping's industry contribution seems insignificant compared to other sectors, it is worth noting that the emissions have seen an increase of 9.6% compared to 2012 primarily due to the growth in demand for global trade; therefore the International Maritime Organization (IMO) has enforced a strategy for reducing the total GHC emissions by at least 40% by 2030, pursuing efforts towards 70% by 2050, compared to 2008, through a series of mandatory regulations.

These regulations were introduced in MARPOL Annex VI, and the main subject is the efficiency of the new ships defined by the energy efficiency design index (EEDI), divided into various phases with stricter requirements for the maximum value of EEDI to achieve a smooth transition. In essence, EEDI reflects the effectiveness of the design as a whole, by accounting for the installed power, attained speed and payload. Since the environmental effort for reducing the emissions is a collective

endeavour among the shipping industry, the existing vessels constitute no exemption; hence the energy efficiency existing index (EEXI) and carbon intensity indicator (CII) came into force. EEXI is inherently the EEDI but altered to be applicable for the existing vessels, while the CII is an index assessing the operational efficiency by measuring the actual annual fuel consumption with respect to the travelled distance.

In that context, the proposed regulations encourage the naval architects and ship owners to actively cooperate in searching for environmentally friendly solutions. For instance, hull shape and power plant optimization have been proposed, but these do not apply to the existing vessels. Thus, an alternative would be operational improvements, such as weather routing, the process of choosing the optimal route based on the weather forecast, slow steaming, and regularly cleaning the hull to reduce the viscous resistance created by attached sea organisms. An apparent yet efficacious approach is trim optimization, which inspired the subject of this thesis and can easily be implemented by changing the load distribution. Any improvement, even small ones, gained by utilizing those techniques is crucial to ships' operation as they reduce fuel consumption and ensure compliance with the regulation of IMO.

## 1.2 Literature Review

Trim optimization is a useful tool in the disposal of shipowners for significantly reducing fuel consumption and operational costs, as 5% less power is required between the optimum trim and even keel for deep water [6]. The optimum value is highly dependent on the speed and depth of the vessel; thus this decrease may vary unless an extensive study is conducted to find the optimum trim for different scenarios. Various techniques have been proposed throughout the previous years, the most obvious being the experimental investigation in a towing tank, as it is the primary method available to the marine engineer for understanding complex hydrodynamic phenomena. In [11], a thorough study was conducted on trim's influence for six different ship models in several loading conditions.

As technology evolves and more powerful computers become commercially available, numerical methods are becoming more and more popular, and the problem of trim optimization is no exemption. In that respect, several studies have been completed by utilizing Computational Fluid Dynamics (CFD) methods. Part of the transition towards the numerical approach is the data validation between the experimental fluid dynamics (EFD) and CFD to assess the effectiveness of the method, and per [26] power savings gained are similar between the two techniques. However, when performing those comparisons, the importance of the scale effects should not be ignored, particularly when investigating propulsion-improving devices and trim

optimization [10], because of the difference in the size of the boundary layer, the damping of the vortices and the relationship among the viscous flow and the wave patterns. Also, the time-consuming process of model tests combined with the high cost of manufacturing the models makes the CFD approach more advantageous, notably when solving for the full ship size. Also, towards this direction contributed the ITTC, which published guidelines for CFD applications [25] and created a standard for the numerical simulations to properly assess the results and the uncertainty involved for ensuring safe comparisons. Until this point, the researches were theoretically investigating the ship performance for several trim angles and sinkage, however [35] extended even further the concept by examining the application on a real ship by developing a program for optimizing the trim based on the engine power data of the actual vessel, derived from the simulations, with the response surface method (RSM) for selecting the best value, and then applying the results on real's ship journey. The experiment concluded that the energy savings achieved in a year are approximately 5-8%, which accounts for 640 tonnes of fuel for 200 days of operation.

In [18] an automated numerical process was developed for analyzing the trim's effect on the ship's performance and compared the results with model tests, giving a good correlation. An added complexity is present in this study, as the effects of the propeller are also integrated into the simulation providing a more realistic approach. Despite being a simple idea, the automation process of the simulation setup and the post process of the data through the SoLuTion system provided by Samsung Heavy Industries Co. (SHI) by predefining a file with the data for speed, draught and trim is a powerful and efficient tool which saves valuable time.

Potential flow has also been examined as an alternative solution to the RANS equations of the CFD approach due to less computational cost. The main focus of the study in [20] was the use of a potential panel method for predicting the optimum trim. The results showed less resistance for the trimmed condition, however between the Wigley and HUST hulls examined only the first had a good agreement with the experimental data for all the scenarios, while the latter had some inconsistencies among the different displacements. Regardless of mixed conclusions, the method is assumed to be valid for tackling the problem of trim optimization. In [19], the Taylor Expansion Boundary Element Method (TEBEM) was utilized, for calculating the added resistance in regular waves for the optimization problem, while the irregular waves were estimated with ITTC's two-parameter spectrum. Through the analysis, a dynamic trim was proposed for a specific route by modelling the sea state impact and concluded in a 0.04% potential fuel saving compared to the calm water resistance trim optimization.

Following the current trends [37] gave an interesting perspective on the trim optimization problem by suggesting a method based on machine learning. Essentially, existing experiments for six containerships were used to get their characteristic parameters, which then trained the machine learning model and created a method to get any containerships' optimum trim. Additionally, the methodology was investigated for a 4250 TEU containership showing consistency with the experimental data.

The impact of the bulbous shape and immersion is significant in the effective establishment of the optimum trim for a vessel because of its impact on the total resistance through the wave-making resistance. In that respect, the hull form optimization is regarded, as an extension of the trim optimization, but with applicability mainly on new building vessels, promising resistance reduction. Several studies have been completed investigating these subjects and fall into two categories based on the approach of geometry parametrization, generative and deformative. The first approach requires the parametrization of the baseline geometry with CAD software, while the latter parametrizes only the changes in the initial geometry. A comparison of both approaches was given by [36], in which two different optimization methods, based on the geometry handling, were developed and compared, yielding similar results. However, the deformation approach is more compelling because it eliminates the need for additional software, which in most cases is not open-source.

## 1.3 Scope of the thesis

The motivation behind this thesis was to establish an automated process for calculating the optimum trim of a vessel by coupling two independent open source softwares.

The study encompasses the following key aspects:

- The research involves the use of OpenFOAM, a CFD software, to simulate and analyze the hydrodynamic behaviour of the vessel and includes the modeling of the flow around the hull for determining the forces and moments acting on the vessel.
- The investigation incorporates the integration of OpenMDAO, an optimization framework implemented in Python, to calculate the optimum value for the trim.
- The study aims to develop a coupling methodology between the aforementioned softwares, allowing for seamless data exchange and iterative optimization. The focus will be on achieving a bi-directional coupling, enabling the

transfer of calm water resistance from OpenFOAM to OpenMDAO and incorporating the resulting trim angle back into the hydrodynamic analysis.

- This study will primarily focus on the KCS hull geometry and will concentrate on trim optimization at a single operating condition. Additionally, a single optimization algorithm will be used to efficiently search for the optimal trim angle within specified constrains.

- Investigate the extension of the proposed methodology for the bulbous shape optimization.

## 1.4 Thesis Outline

- Chapter 2: The analytical equations describing the flow past an immersed body are discussed; however, it is apparent that in this form a solution cannot be extracted. Therefore, the method for converting the equations into a set of numerical ones is presented.

- Chapter 3: The process for calculating calm water resistance in the OpenFOAM environment is shown. First, by describing the geometry used along with the generation of the numerical towing tank and the corresponding boundary conditions. The chapter is concluded with the mesh sensitivity analysis and the final results of the simulation for the chosen mesh resolution.

- Chapter 4 The establishment of the numerical simulation is followed by the main topic of interest, trim optimization. The framework for the trim optimization process is presented through a flowchart and the chosen optimization algorithm is briefly explained. Moreover, a combination of trim and sinkage optimization is also discussed to demonstrate the applicability of the framework with two design variables.

- Chapter 5: The successful implementation of the optimization process motivated an extension by investigating the shape optimization of the bulbous. Thus, for proving the concept a methodology for altering the geometry along with the corresponding theory is presented. Finally, the coupling with the OpenFOAM and the optimizer followed by the results denote the closure of the chapter and this work.

- Chapter 6: The work discussed in the previous chapters is summarized and proposals for future work are suggested.

## 1.5 Ship's resistance

The ship resistance for a specific constant speed is defined as the force required to tow the vessel in calm water and can be further categorized into the bare hull or propulsion resistance, whether or not the appendages are accounted for. The resistance is the product of the interaction between the immersed hull and the surrounding fluid since a force is acting on the hull which can be analyzed into a normal and tangential part caused by pressure and viscosity respectively. Therefore, the total resistance is the projection in the direction of speed of these forces integral on the surface of the hull. Hence, the friction $R_F$ and pressure $R_R$ resistance components emerge, denoting the integration of the shear (viscous) and normal (pressure) stresses on the hull surface and projecting them in the direction of the ship's motion, accordingly.

The total resistance is then defined as the sum of friction and pressure resistance, however, as shown in 1.1, several other components comprise the ship's resistance that cannot explicitly be quantified.

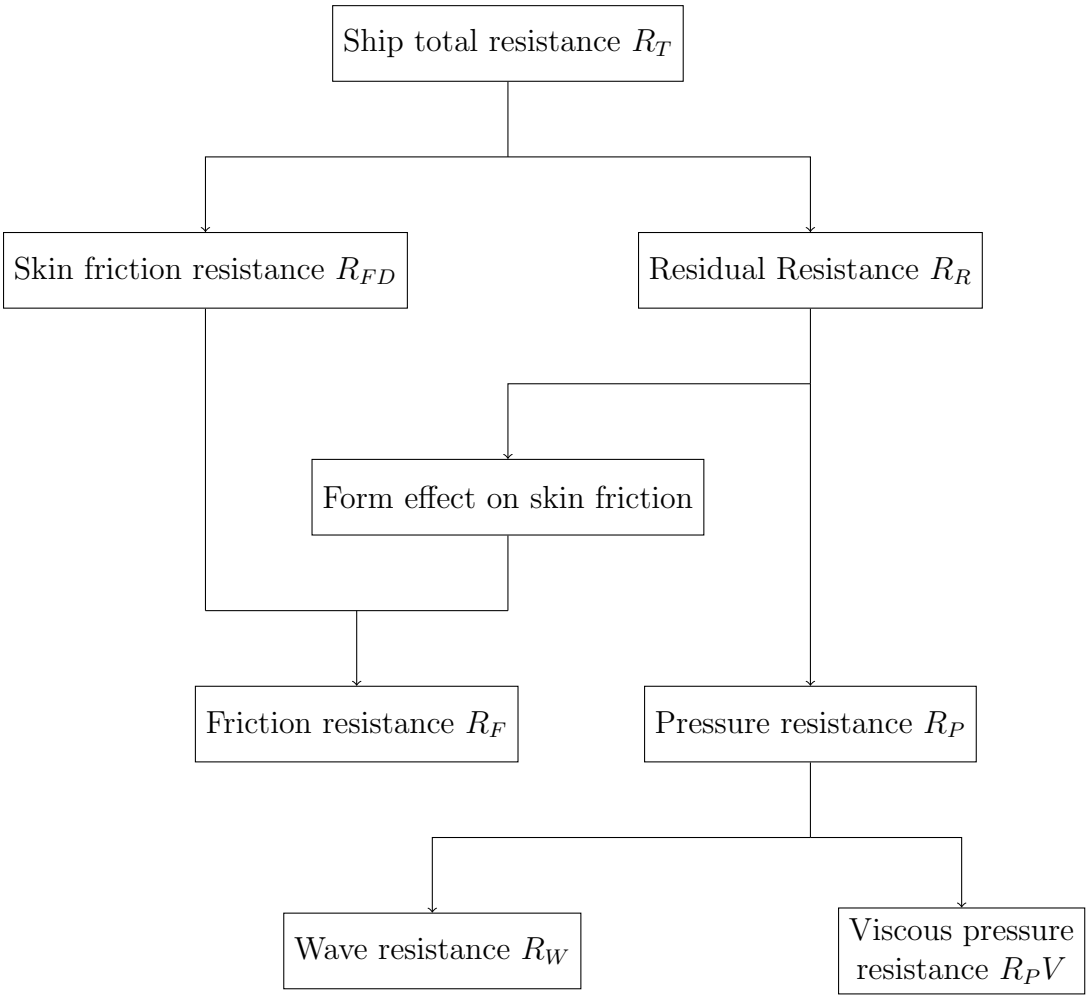



**Figure 1.1:** *Resistance Components*

## Friction Resistance

Friction resistance is generated by the tangential stresses acting on the wetted surface of a body moving through a fluid medium. These stresses arise due to the molecular interactions between the fluid and the body's surface. Frictional resistance is observed in viscous fluids that exhibit cohesive behaviour and is closely associated with the shear stresses that develop between the fluid and the body. The magnitude of frictional resistance is influenced by the Reynolds number, which characterizes the flow regime. During the movement of a vessel, the fluid particles in immediate contact with the body's surface have the same velocity as the moving body, adhering to the "no-slip" condition. However, as the distance from the body increases, the fluid particles remain unaffected by the body's motion. The velocity of the fluid experiences a significant reduction as it progresses from the body towards the lateral sides. This deceleration occurs primarily within a thin layer adjacent to the body known as the boundary layer. In the bow region, the boundary layer is relatively thin, while it gradually thickens as it progresses towards the stern, as a result the geometry that the fluid "sees" changes. The behaviour of the boundary layer and the associated changes in fluid velocity play a crucial role in determining the overall frictional resistance experienced by the moving body. Understanding and managing the boundary layer characteristics is essential for optimizing the performance and efficiency of fluid-flow-related systems, such as ships.

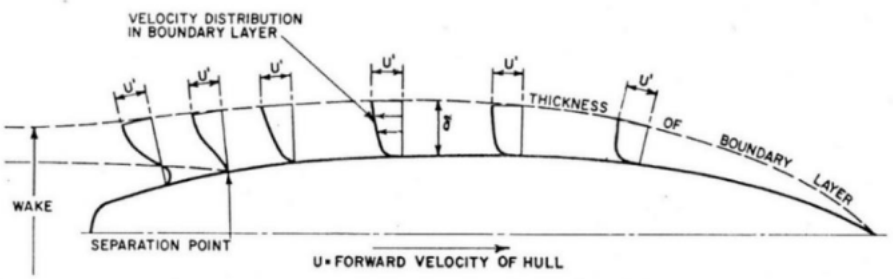



**Figure 1.2:** *Schematic diagram of boundary layer flow.*

## Pressure Resistance

The other component contributing to the total resistance is the pressure resistance, which is caused by normal pressures acting on the wetted surface independently of viscosity and is dependent on the molecular motion of the fluid. This resistance is related to the Froude number, which characterizes the importance of inertia forces compared to gravitational forces. The main constituents of pressure resistance are the viscous pressure resistance and the wave resistance. The viscous pressure resistance arises from the presence of cohesive forces within the fluid and is influenced by the properties of the fluid. The wave resistance, on the other hand, is associated

with the generation and propagation of waves as the body moves through the fluid. The magnitude of pressure resistance is controlled by various factors such as the shape and geometry of the body, the speed of the body through the fluid and the surrounding fluid.

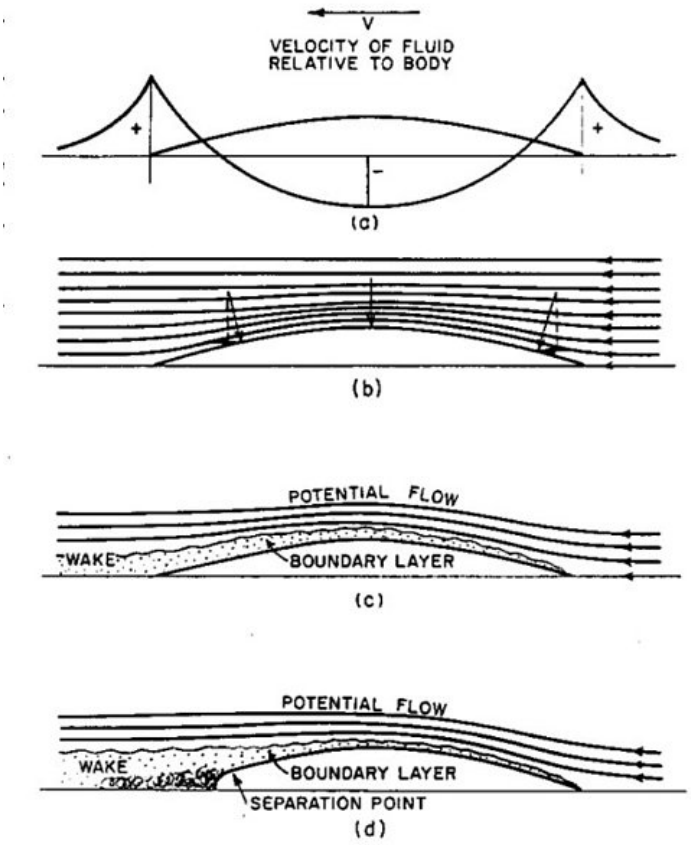



**Figure 1.3:** *Fluid flow around immersed body.*

### Viscous pressure resistance

The viscous pressure resistance can be calculated through the integration of the normal stresses and is caused due to the viscosity of the fluid and the created boundary layer. Moreover, in the case of a fully submerged body in an inviscid fluid, as shown in figure 1.3 (b), the friction resistance is equal to zero, as the wave resistance, because of the absence of the free surface. Also, from the same figure, it is evident that the pressure follows the geometry of the hull, and the flow accelerates as it travels from the fore to the aft body; by applying the Bernoulli equation the maximum speed value is located in the parallel mid-body, while in the stern and stem the velocity is equal to zero (stagnation points). The pressure distribution is, therefore, as shown in figure 1.3 (a). From the previous analysis, it is concluded that the forces exist due to pressure stresses, but since they are normal to the hull, they cancel each other resulting in zero resistance.

However, for a submerged body in a viscous fluid, as explained in friction resistance, a boundary layer is created and alters the geometry; hence the pressure distribution along the body differs in contrast to the inviscid flow, and the stresses in the stern decrease. Therefore, the pressure's lateral component reduces while the viscous pressure resistance increases.

#### Wave resistance

Ships operate partially immersed in water, resulting in the presence of a free surface that contributes to wave resistance, denoted as $R_W$. Wave resistance represents the energy absorbed by the generated waves during the ship's motion. Lord Kelvin was one of the first researchers to study this phenomenon and proposed that a moving ship has a single pressure point that travels along the direction of motion, maintaining a constant value. From this pressure point, waves radiate, forming a distinct wave pattern consisting of transverse and divergent waves. Figure 1.4 illustrates this pattern, where two straight lines originating from the pressure point enclose the wave system at an angle of 19.5 deg relative to the ship's speed vector. The specific form of these waves is unique to each vessel and heavily influenced by the hull shape. Notably, the presence of a bulbous bow, if incorporated, can significantly impact wave resistance as it introduces its own wave system that can partially counteract the effect of Kelvin waves.

## 1.6 OpenFOAM software

The OpenFOAM (**Open** Source **F**ield **O**peration **A** and **M**anipulation) software was utilized for the required simulations in this thesis. It is an open-source software for computational fluid mechanics (CFD) and numerous other applications even in the field of economics. The code was initially developed at the Imperial College by Henry Weller [38] in the late 80s but it was released in 2004. The software is written in the C++ programming language, and its primary function is to create executable files, known as applications. These applications are divided into two categories: solvers, which are designed to solve specific problems, and various utilities that perform tasks such as generating and controlling the grid, transforming data for use by other software packages, and processing the results obtained from problem-solving. OpenFOAM provides also the ability for running applications in parallel, also referred to as domain decomposition, in which the domain is divided into pieces and allocated to separate processors. Parallel execution employs the openMPI implementation of the widely-used message-passing interface (MPI), which is available in the public domain. It is a compelling tool for research applications because custom

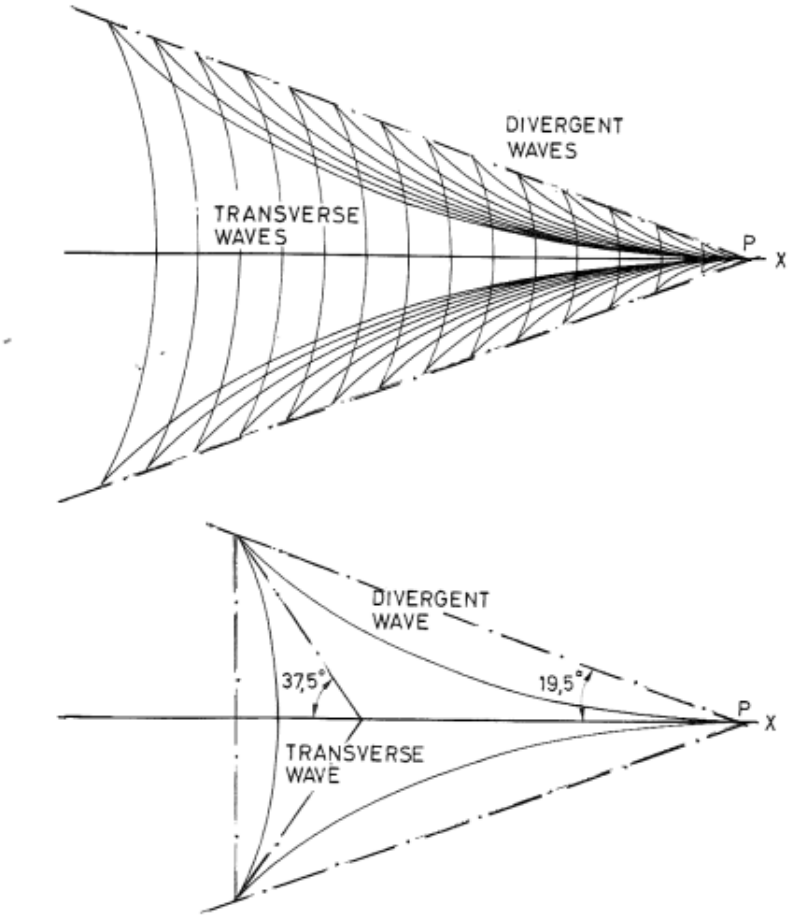



**Figure 1.4:** *Kelvin wave pattern.*

projects can be developed and integrated into the existing code. Also, OpenFOAM can be seamlessly coupled with other softwares, since all its commands are called through the terminal.

## 1.7 OpenMDAO

OpenMDAO stands for Open Multidisciplinary Design Analysis and Optimization and is an open-source optimization framework written in Python and was developed by NASA Glenn Research Center [4] to address complex aircraft design optimization problems. OpenMDAO framework comprises a collection of Python packages designed to simplify the creation and execution of analysis models. It is highly resilient, as it can accommodate any analysis software that can be executed through a command line interface and utilizes input and output files that are parseable. The tools provided by OpenMDAO aid in the development process, making it easier to work with various analysis software systems. It can efficiently and accurately solve complex model derivates, as it is primarily designed for handling gradient-based optimization; however, numerous features are available for handling gradient-free

optimization problems as well. The OpenMDAO framework is developed to assist in the integration of separate software components to perform combined analyses, and it can work both in serial and parallel computing environments. Finally, OpenMDAO provides a comprehensive suite of optimization algorithms and techniques to find optimal solutions to their design problems. These optimization capabilities can be applied to both single-discipline and multi-discipline problems, enabling users to explore trade-offs and find designs that meet various constraints and objectives.

# Chapter 2

# Governing Equations

## 2.1 Introduction

In this chapter, the governing equations for an external fluid flow will be presented, along with the difficulties arising in solving them. The forces acting upon a ship are caused due to its interaction with the surrounding water and the free surface. Therefore, the governing equations are presented to get an understanding of the physical phenomenon occurring. They consist of the continuity and linear momentum equations; however, the type of flow should be examined to determine the extent of the applicability of these equations. Since the examined problem is turbulent and a numerical approach was selected for calculating the calm water resistance, the original governing equations are transformed accordingly, so that the numerical computation can be achieved. Afterwards, the modelling of the turbulence along with a methodology for capturing the free surface via the Volume of Fluid Method (VOF) are discussed. Finally, the resulting linear system occurring from the RANS equations is briefly described and the PISO algorithm that is responsible for numerically solving the equations is presented.

## 2.2 Navier-Stokes Equations for incompressible fluid

The ultimate goal of the Computational Fluid Dynamics is to depict the fluid flow as accurately as possible, and this is achieved by utilizing the Navier-Stokes (NS) equations accordingly. These are highly nonlinear second order partial differential equations in four independent variables and describe the derivations of the continuity, momentum and energy equations. As mentioned above, water is incompressible, so the energy equation is ignored in the context of this study.

### 2.2.1 Continuity Equation

The continuity equation or conservation of mass states that the fluid mass cannot change, in the absence of mass sources and sinks. The formula of the equation according to classical mechanics in the differential form is:

$$\frac{\partial \varrho}{\partial t} + \nabla(\varrho \mathbf{u}) = 0 \tag{2.1}$$

where $\mathbf{u} = (u, v, w)$ is the velocity vector of the flow and $\mathbf{t}$ time.

For an incompressible fluid the density is constant over time ($\frac{\partial \varrho}{\partial t} = 0$), thus the continuity equation is as follows

$$\nabla \mathbf{u} = 0 \Leftrightarrow \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} = 0 \Rightarrow R^p = \frac{\partial u_i}{\partial x_i} = 0 \tag{2.2}$$

where $\mathbf{R^p}$ is the residual of the continuity equation and $\mathbf{i} = 1, 2, 3$ denotes the coordinate directions x, y, z respectively.

### 2.2.2 Conservation of Linear Momentum

The conservation of momentum is derived from Newton's second law of motion, and states that the external forces acting on the fluid are equal to the sum of the time variation of the fluid's momentum and the momentum entering and exiting the flow field. Therefore, the above-mentioned law can be formulated mathematically as follows:

$$\frac{\mathrm{d}\mathbf{J}}{\mathrm{d}t} + \mathbf{S} = \sum_{i=1}^{n} \mathbf{F}_i \tag{2.3}$$

$$\frac{\partial u_i}{\partial t} + u_j \frac{\partial u_i}{\partial x_j} = -\frac{\partial p}{\partial x_i} + \frac{\partial}{\partial x_j}[\nu(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i})] \tag{2.4}$$

where $\mathbf{J}$ is the momentum of the fluid, $\mathbf{S}$ is momentum flux, $\sum_{i=1}^{n} \mathbf{F}_i$ is the sum of the external forces and $\nu$ is the kinematic viscocity. In equation 2.4 the symbol $p$ is the pressure divided by the fluid density.

## 2.3 Turbulence

The conservation of mass and linear momentum equations, as presented in the previous chapters form a set of coupled, non-linear partial differential equations. To

determine whether an analytical solution exists or not, it is necessary to examine the type of flow involved in the problem of ship resistance. Most fluids in nature are viscous, meaning that they exhibit resistance due to internal friction between fluid particles. The extent of the viscous behavior is described by the dimensionless Reynolds number:

$$Re = \frac{VL}{\nu} \tag{2.5}$$

where $V$ and $L$ are the characteristic velocity and length of scales of the fluid and $\nu$ is the kinematic viscosity. For small values of the Reynolds number, the effect is controlled by the viscous forces, so the inertial forces can be neglected, while for large values the inertial forces are dominant, indicating turbelent flow. The transition from laminar to turbulent flow, which occurs for moderate Reynold numbers between $Re_{critical} = 10^5 \sim 10^6$ [39], is a complex phenomenon that has yet to be modelled.

In the following chapters, the KCS model is studied for the development of an optimization framework. Given the ship's dimensions and speed, the model has a $Re = 14.03 \times 10^6$ from 3.1 and therefore the flow is turbulent. As a result, eddies of all sizes, with chaotic behavior, are being created because the velocity field changes erratically in every given moment and position.

A solution to the NS equations must be determined now that the characteristics of the flow are known. Although they accurately describe the fluid flow in great detail, the presence of turbulence is inherently the reason why an analytical solution does not exist. An obvious way would be a direct approach to the problem as proposed by the Direct Numerical Simulation, but this would require a huge amount of processing power as all the scales of the turbulence are solved. Due to the rapid development in computing technology in the previous decades, there have been examples of Direct Numerical Simulations (DNS). However, even now these solutions are applied to small and moderate values of the Re number since the computing power required increases dramatically for high turbulent flows ($Re^3$) [24]. Therefore, this method is not applicable in this thesis and another way to tackle the problem should be applied.

For highly turbulent external flows, such as the estimation of ship resistance, the time-averaged turbulent flow field is investigated, because the fluctuations of the acting forces are relatively small. The method developed for solving the time average motion is the Reynolds Averaged Navier Stokes Equations (RANS).

### 2.3.1 RANS Equations

The idea behind the Reynolds averaged method is the decomposition of the flow's variables. Hence, a time-mean value and a fluctuating value for the corresponding variable occur. Therefore, for any given variable $\phi(\mathbf{x}, t)$, where $\phi$ represents the flow variables, it is

$$\phi(\mathbf{x}, t) = \bar{\phi}(\mathbf{x}, t) + \phi'(\mathbf{x}, t) \tag{2.6}$$

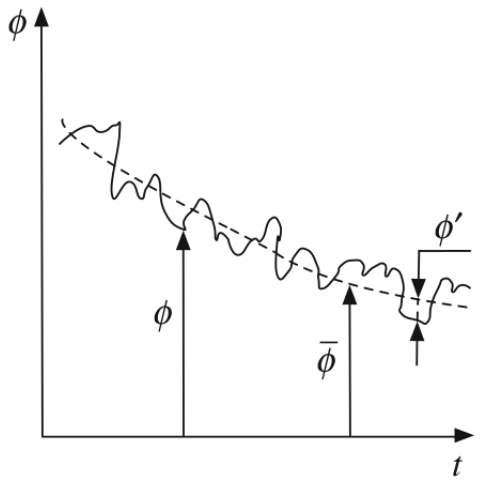



**Figure 2.1:** *Flow variable and is corresponding components $\bar{\phi}$ and $\phi'$*

By utilizing the method of decomposition, the conservation of linear momentum for an incompressible fluid becomes [1],

$$\bar{u_j}\frac{\partial \bar{u_i}}{\partial x_j} = \frac{\partial}{\partial x_j}[-\bar{p}\delta_{ij} + \nu(\frac{\partial \bar{u_i}}{\partial x_j} + \frac{\partial \bar{u_j}}{\partial x_i}) - \overline{u_i' u_j'}] \tag{2.7}$$

where, $\delta_{ij} = \begin{cases} 1, i = j \\ 0, i \neq j \end{cases}$, is the Kroneker's equation.

By comparing the equations 2.3 and 2.7 it is evident that the non-linear term $\overline{u_i' u_j'}$, appears. This term is called Reynolds Stress Tensor and is consequently an extra unknown variable that needs to be determined. A solution was given by Boussinesq's hypothesis, in which he proposed the following formulation for the calculation of the stress tensor [30]:

$$-\overline{u_i' u_j'} = \mu_t(\frac{\partial \bar{u_i}}{\partial x_j} + \frac{\partial \bar{u_j}}{\partial x_i}) - \frac{2}{3}k\delta_{ij} \tag{2.8}$$

where, $\mu_t$ is the turbulent eddy viscosity and $k = \frac{1}{2}\overline{u_i' u_j'}$ is the turbulent kinetic

energy.

Therefore, a way to calculate the turbulent eddy viscosity is essential, in order to fully define the stress tensor. As already mentioned the turbulence is in a sense random, thus various methods, known as turbulence models, have been proposed to predict the eddy viscosity.

### 2.3.2 Turbulence Modelling

By utilizing the time averaging for the NS equations, as previously presented, more unknown variables occur than the equations available, resulting in an unsolvable problem. However, a closure to this problem is given by the various eddy viscosity models that have been developed based on Boussinesq's hypothesis. These models are categorized as follows:

- Algebraic or zero equation models: as per the name these models use algebraic equations for the calculation of the turbulent eddy viscosity
- One equation models: the turbulent eddy viscosity is computed by solving one differential equation
- Two equation models: the turbulent eddy viscosity is computed by solving two differential equations
- Second order closure models:

In the current thesis, the $k-\omega$ SST model will be used for the turbulence modeling, since it is the combination of the $k-\varepsilon$ and $k-\omega$ models, those should be presented as well.

### 2.3.3 The $k-\varepsilon$ model

The $k-\varepsilon$ model [17] is among the most used eddy viscosity models in the field of CFD and it falls into the two equation models category. The turbulent kinetic energy k and the turbulent energy dissipation $\varepsilon$ rate, as presented in this model are essentially describing the behavior of the turbulence. Therefore two additional transport equations for k and $\varepsilon$, as described below, are computed.

$$\frac{\partial k}{\partial t}+\frac{\partial (k u_i)}{\partial x_i}=\frac{1}{\rho}\frac{\partial}{\partial x_k}\left[\frac{\mu_t}{\sigma_k}\frac{\partial k}{\partial x_k}\right]+\frac{\mu_t}{\rho}\left(\frac{\partial U_i}{\partial x_k}+\frac{\partial U_k}{\partial x_i}\right)\frac{\partial U_i}{\partial x_k}-\varepsilon \tag{2.9}$$

$$\frac{\partial \varepsilon}{\partial t}+\frac{\partial (\varepsilon u_i)}{\partial x_i}=\frac{1}{\rho}\frac{\partial}{\partial x_k}\left[\frac{\mu_t}{\sigma_\varepsilon}\frac{\partial \varepsilon}{\partial x_k}\right]+\frac{C_1\mu_t}{\rho}\frac{\varepsilon}{k}\left(\frac{\partial U_i}{\partial x_k}+\frac{\partial U_k}{\partial x_i}\right)\frac{\partial U_i}{\partial x_k}-C_2\frac{\varepsilon^2}{k} \tag{2.10}$$

where, $C_1 = 1.44$, $C_2 = 1.92$, $\sigma_k = 1.0$, $\sigma_\varepsilon = 1.3$ and $\mu_t = C_\mu \frac{k^2}{\varepsilon}$, with $C_\mu = 0.09$. Also, the term $U_i$ corresponds to the mean component of velocity in direction $x_i$.

However, some limitations apply to this proposed model since the equations cannot be integrated near the wall. Also, the $k-\varepsilon$ model is only applicable to fully developed turbulent flows. As a result, further studies have been conducted for the development of new models in order to overcome those limitations.

### 2.3.4 The $k-\omega$ model

To resolve the problematic behavior of the $k-\varepsilon$ model close to the wall, the $k-\omega$ model was developed in which the turbulent energy dissipation $\varepsilon$ is replaced by the rate at which turbulence kinetic energy is converted into internal thermal energy $\omega$. The specific dissipation ($\omega$) determines the scale of turbulence. Those two models share quite a few similarities since both of them require the computation of two transport equations and are based on Boussinesq's hypothesis. The extra two PDEs of this model are as follows [40]

$$\frac{\partial(\rho k)}{\partial t} + \frac{\partial(\rho u_j k)}{\partial x_j} = \tau_{ij}\frac{\partial u_i}{\partial x_j} - \beta^*\rho\omega k + \frac{\partial}{\partial x_j}\left[\left(\mu + \sigma_k \frac{\rho k}{\omega}\right)\frac{\partial k}{\partial x_j}\right] \tag{2.11}$$

$$\frac{\partial(\rho \omega)}{\partial t} + \frac{\partial(\rho u_j \omega)}{\partial x_j} = \frac{\gamma\omega}{k}\tau_{ij}\frac{\partial u_i}{\partial x_j} - \beta\rho\omega^2 + \frac{\partial}{\partial x_j}\left[\left(\mu + \sigma_\omega \frac{\rho k}{\omega}\right)\frac{\partial \omega}{\partial x_j}\right] \tag{2.12}$$

where, $\mu_t = \frac{\rho k}{\omega}$ the turbulent eddy viscosity, $\tau_{ij}$ is the Reynolds stress tensor, $\sigma_k = 0.5$, $\sigma_\omega = 0.5$, $\beta^* = 0.09$ and $\gamma = 5/9$.

The $k-\omega$ model performs better than the $k-\varepsilon$ due to its superior behaviour in handling flows close to the wall and weak adverse pressure gradients, as well as the ease of solving the integrals. Despite all these advantages its inability to handle the free stream imposes a significant weakness, particularly in external flow problems like calculating the ship resistance.

### 2.3.5 The $k-\omega$ SST model

The $k-\omega$ Shear Stress Transport model is a two equation eddy viscosity model which combines the advantages of the two previously discussed models to overcome their weaknesses. Therefore, the $k-\omega$ SST model can be applied close to the wall without the need of damping and the free stream is not sensitive to the designation of the free stream. Essentially, it is a hybrid model which switches between $k-\varepsilon$ and $k-\omega$ according to the region applied.

The two equations for the kinetic energy $k$ and the specific dissipation rate $\omega$ which govern the model are [21]

$$\frac{\partial(\rho k)}{\partial t} + \frac{\partial(\rho U_i k)}{\partial x_i} = P_k - \beta^* \rho k \omega + \frac{\partial}{\partial x_i}\left[(\mu + \sigma_k \mu_t)\frac{\partial k}{\partial x_i}\right] \tag{2.13}$$

$$\frac{\partial(\rho\omega)}{\partial t} + \frac{\partial(\rho U_i \omega)}{\partial x_i} = \alpha\rho S^2 - \beta\rho\omega^2 + \frac{\partial}{\partial x_i}\left[(\mu + \sigma_\omega \mu_t)\frac{\partial \omega}{\partial x_i}\right] + 2(1 - F_1)\rho\sigma_{\omega 2}\frac{1}{\omega}\frac{\partial k}{\partial x_i}\frac{\partial \omega}{\partial x_i} \tag{2.14}$$

where $P_k$ is a production limiter of the turbulent kinetic energy and is equal to

$$P_k = \min\left(\mu_t \frac{\partial U_i}{\partial x_j}\left(\frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i}\right), 10\beta^*\rho k\omega\right)$$

The eddy viscosity $\mu_t$ is

$$\mu_t = \frac{\alpha_1 k}{\max(\alpha_1\omega, SF_2)}$$

where $S = \sqrt{2S_{ij}S_{ij}}$ is the invariant measure of the strain rate and the functions $F_1$ and $F_2$ are the blending functions of the model and whem $F_1 = 0$ the $k - \varepsilon$ is applied, while the $k - \omega$ is employed when $F_1 = 1$. The formulation of those are as follow

$$F_1 = \tanh\left\{\left\{\min\left[\max\left(\frac{\sqrt{k}}{\beta^*\omega y}, \frac{500\nu}{y^2\omega}\right), \frac{4\rho\sigma_{\omega 2k}}{CD_{k\omega}y^2}\right]\right\}^4\right\} \tag{2.15}$$

where y is the distance to the nearest wall and

$$CD_{k\omega} = \max\left(2\rho\sigma_{\omega 2}\frac{1}{\omega}\frac{\partial k}{\partial x_i}\frac{\partial \omega}{\partial x_i}, 10^{-10}\right) \tag{2.16}$$

$$F_2 = \tanh\left\{\left[\max\left(\frac{2\sqrt{k}}{\beta^*\omega y}, \frac{500\nu}{y^2\omega}\right)\right]^2\right\} \tag{2.17}$$

The constants subject to the model are $\beta^* = 0.09$, $\alpha_1 = 5/9$, $\beta_1 = 0.075$, $\sigma_{k1} = 0.85$, $\sigma_{\omega 1} = 0.5$, $\alpha_2 = 0.44$, $\beta_2 = 0.0828$, $\sigma_{k2} = 1.0$ and $\sigma_{\omega 2} = 0.856$. The blending occurs in this model also affects those constants which are further computed as $\alpha = \alpha_1 F + \alpha_2(1 - F)$.

## 2.4 Volume of Fluid (VOF) Method

The problem of calculating the ship resistance in calm water is a two-phase unsteady flow consisting of two immiscible fluids. Thus, there is a need for developing a methodology that takes into consideration the phenomena occurring on the free surface. Numerous techniques have been employed in the past to estimate free boundaries in finite difference numerical simulations. A straightforward yet effective approach based on the idea of utilizing a fractional volume of fluid (VOF) [9] will be implemented in the present thesis. Therefore, the free surface is solved through a single scalar function $\alpha$ called volume fraction, whose values range between 0 and 1 for water and air respectively.

The properties of the fluid mixture can now be defined with the help of the volume fraction as per below.

$$\rho_m = \alpha\rho_w + (1-\alpha)\rho_\alpha \tag{2.18}$$

$$\mu_m = \alpha\mu_w + (1-\alpha)\mu_\alpha \tag{2.19}$$

where $\rho$ is density, $\mu$ is the viscosity and the symbols $\alpha$, w, m corresponds to the air, water and the cell to be resolved respectively.

Therefore, the scalar function is calculated for the whole computational mesh through the following PDE which is then coupled with the conservation of mass and momentum.

$$\frac{\partial\alpha}{\partial t} + \nabla(\mathbf{u}\alpha) = 0 \tag{2.20}$$

The transition of the scalar function from 0 to 1 defines the interface of the fluids. A discontinuity is created between the two fluids by the abrupt change occurring in the interface and subsequently, the gradient of the scalar function becomes difficult to resolve. A solution is given by the modification of the original PDE [29].

$$\frac{\partial\alpha}{\partial t} + \nabla(\alpha\mathbf{u}) + \nabla(\alpha(1-\alpha)\mathbf{u_r}) = 0 \tag{2.21}$$

where $\mathbf{u_r}$ is the artificial velocity field.

## 2.5 Numerical Solution of the Governing Equations

The previous chapters introduced the equations governing the flow past immersed bodies. The analytical solution is only applicable to a small range of engineering problems, and the calculation of ship resistance is no exception. Therefore, the adoption of a numerical solution is necessary. For this purpose, the cell centered Finite Volume Method (FVM) is implemented over an unstructured mesh. Using this technique, the NS equations can now be solved as they are transformed into a system of linearised algebraic equations. The final step of the process is the implementation of the PISO algorithm [13] to reach the desired solution.

### 2.5.1 PISO Algorithm

Even the numerical approach of the NS equations is a complex task primarily because of the pressure's presence in all the momentum equations. A technic to overcome the difficulties imposed is initially to predict the pressure field in such a way that the continuity equation is satisfied. This method is called pressure velocity coupling and in this category falls also the PISO algorithm used in the current thesis, which is an extension of the widely used SIMPLE algorithm [23]. The selection of this specific algorithm was based on its ability to solve unsteady flow simulations while being time effective.

Essentially, the variables to be determined are the velocity and pressure at a time $t_{n+1}$ while knowing the values of the current time step ($t_n$). In the momentum equations non-linear terms exist, which are then converted to linear with the help of discritization schemes and solved. Therefore, by grouping together all the the terms the following linear system occurs [22]:

$$\mathbf{u}_C + \mathbf{H_C}[\mathbf{u}] = -\mathbf{D}_C^{\mathbf{u}} \left(\nabla p\right)_C + \mathbf{B}_C^{\mathbf{u}} \tag{2.22}$$

where $C$ is the cell center, $\mathbf{B}_C^{\mathbf{u}}$ defines the source terms that might be present. The process begins by iteratively solving the linear system defined in 2.22 to get the velocity field $u^*$ while the pressure field is taken from the previous iteration, defined with the superscript $n$. Therefore, the following system occurs.

$$\mathbf{u}_C^* + \mathbf{H}_C[\mathbf{u}^*] = -\mathbf{D}_C^{\mathbf{u}} \left(\nabla p^{(n)}\right)_C + \mathbf{B}_C^{\mathbf{u}} \tag{2.23}$$

However, the obtained velocity field $u^*$ satisfies only the momentum equation and

not the continuity. Thus, a correction for the velocity, mass flow rate and pressure values is implemented, as defined in equation 2.24 below.

$$\begin{aligned} \mathbf{u} &= \mathbf{u}^{*} + \mathbf{u}^{'} \\ p &= p^{(n)} + p^{'} \\ m &= m^{*} + m^{'} \end{aligned} \tag{2.24}$$

The final form of the continuity equation is:

$$\sum_{f \sim nb(C)} \dot{m}_{f}^{'} = - \sum_{f \sim nb(C)} \dot{m}_{f}^{*} \Leftrightarrow \sum_{f \sim nb(C)} \rho_{f} \mathbf{u}_{f}^{'} \mathbf{S}_{f} = - \sum_{f \sim nb(C)} \rho_{f} \mathbf{u}_{f}^{*} \mathbf{S}_{f} \tag{2.25}$$

By subtracting the equations 2.22 and 2.23 the corrected quantities can be calculated.

$$\mathbf{u}_{C}^{'} + H_{C}[\mathbf{u}^{'}] = -D_{C}^{\mathbf{u}} \left( \nabla p^{'} \right)_{C} \tag{2.26}$$

The PISO algorithm can be summarized into the following steps:

1. Solve the transport equation for the VOF.
2. For the solution at a time $t + dt$, the value for the pressure is taken from the previous time $t$ as an initial guess.
3. The momentum equation is solved for the determination of the new velocity field $\mathbf{u}^{*}$, and the values are calculated at the cell faces using an interpolation scheme.
4. The pressure correction equation is assembled and then solved, to get the $p^{'}$.
5. To satisfy the continuity equation, the pressure and velocity fields and the mass flow are updated accordingly.
6. The momentum equation is explicitily solved, after calculating its coefficients by utilizing the latest velocity and pressure fields.
7. The mass flow rates are updated at the cell faces and the pressure correction equation is constructed and solved for the determination of the pressure correction field.
8. The pressure, velocity and mass flow rates fields are updated
9. Return to step 6 and repeat until the number of corrector steps is reached.
10. The turbulence model equation are solved.

11. Proceed to the next time step.
12. Return to step 1 and repeat until the end time is reached.

# Chapter 3

# Numerical Investigation

The goal of this thesis is to optimize the ship's performance by minimizing the resistance. This problem continuously concerns engineers, as the pursuit of a better design solution never ends. In this chapter, the simple problem of numerically predicting the calm water resistance is presented, upon which the trim optimization framework is based in the next chapter. Firstly, the geometry of interest is introduced and afterwards, the way to transform the geometry into a numerical towing tank, by creating an unstructured mesh grid is explained. However, the areas of the computational domain should be defined accordingly to resemble the physical tank; hence the boundary conditions for the simulation initialization are then specified. Afterwards, the settings used for the setup of the computational approach along with a mesh sensitivity analysis for the determination of the most accurate and cost-effective mesh resolution are displayed. The results acquired by the numerical solution complete the chapter.

## 3.1 Ship Geometry

In the context of this study, the KCS geometry will be investigated. It is an experimental bare hull, which has never been built on a full scale, but extensive analysis has been performed both in CFD and wave tanks to assess its performance and characteristics in different scenarios. All this available information and the free access to the actual geometry make the KCS hull the best option for research. In Table 3.1, the main particulars of the hull are displayed.

**Table 3.1:** *KCS Main Particulars*

| Main particulars | Full Scale | Model |
|---|---|---|
| $Scale$ | 1.000 | 31.599 |
| $L_{pp}$ [m] | 230.0 | 7.2786 |
| $L_{wl}$ [m] | 232.5 | 7.3570 |
| $B_{wl}$ [m] | 32.2 | 1.0190 |
| $D$ [m] | 19.0 | 0.6013 |
| $T$ [m] | 10.8 | 0.3418 |
| $Displacement$ $[m^3]$ | 52030 | 1.6490 |
| $S\ w/o\ rudder$ $[m^2]$ | 9530 | 9.5441 |
| $C_B$ | 0.651 | 0.651 |
| $C_M$ | 0.985 | 0.985 |
| $LCB$ [%] | -1.48 | -1.48 |
| $U$ $[m/s]$ | 12.36 | 2.196 |
| $Fn$ | 0.26 | 0.26 |
| $Re \times 10^6$ | 24930.00 | 14.03 |

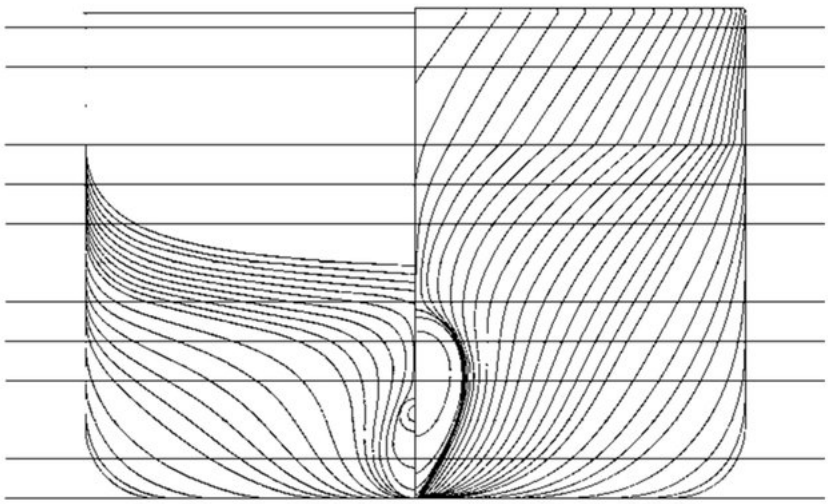


**Figure 3.1:** *The body plan of KCS*

## 3.2 Numerical Towing Tank

The first step in the process of numerically calculating the resistance is to form the volume (numerical wave tank), in which the equations are discretized and subsequently solved. As already mentioned the commercial software OpenFOAM is utilized as it offers numerous tools for the creation of the computational domain. The dimensions should be adequate in all directions in space to capture the physical phenomena occurring due to the interaction between the hull and the free surface.

An unstructured block grid was selected for the representation of the numerical wave tank. The examined problem is a two-phase flow one, thus the domain was divided into three blocks so that the different areas can be independently treated. A box

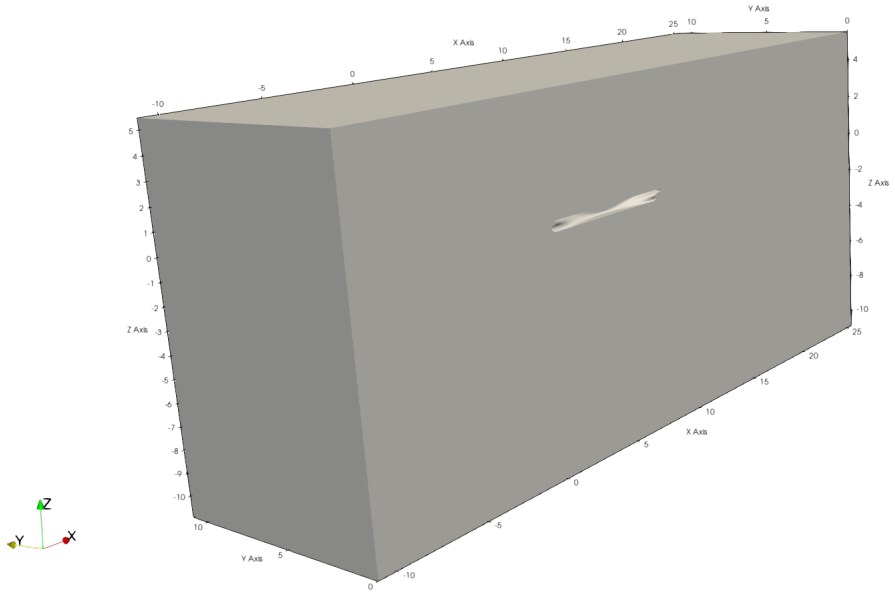



**Figure 3.2:** *Domain*

was placed in such a way to enclose the free surface since it is the main area of interest and the other two were placed below and above defining the bottom and the atmosphere respectively. The boundaries of the domain were split into patches, which are determined by the faces of the created boxes. Also, on those patches, the boundary conditions, for the initialization of the simulation are being applied. The patches are defined by the faces of the boxes consisting of the computational domain and for the convenience of the problem they were named. Thus, as depicted in 3.2 the created patches are the inlet, outlet, bottom, atmosphere, midplane, side denoting the inflow, outflow, the bottom of the wave tank, the top of the tank, the symmetry plane of the ship and the side wall, respectively.

An estimation of the domain size can be given by the main particulars of the vessel and more specifically by the length. To begin with, the length between the stern and the outlet must be adequate to allow the full development of the wake and also the damping of the waves far from the geometry. The area between the inlet and the foremost point of the hull should be spaced in a way to reduce the unnecessary cells but allow for the initialization of the boundary conditions. Finally, the depth is chosen to approximate deep water and the dimension in the y direction to avoid the reflecting waves diverting the simulation results.

The final dimensions of the computational domain are summarized in the following table:

**Table 3.2:** *The final dimensions of the computational domain in meters along with the corresponding cells in each direction.*

| Direction | x | | y | z | |
|---|---|---|---|---|---|
| | Fore | Aft | Side | Down | Up |
| Length/Lpp | 1.5 | 2.5 | 1.5 | 1.5 | 0.75 |
| Length $[m]$ | 10.92 | 18.25 | 10.92 | 10.92 | 5.46 |
| Cells | 25 | | 24 | 84 | |

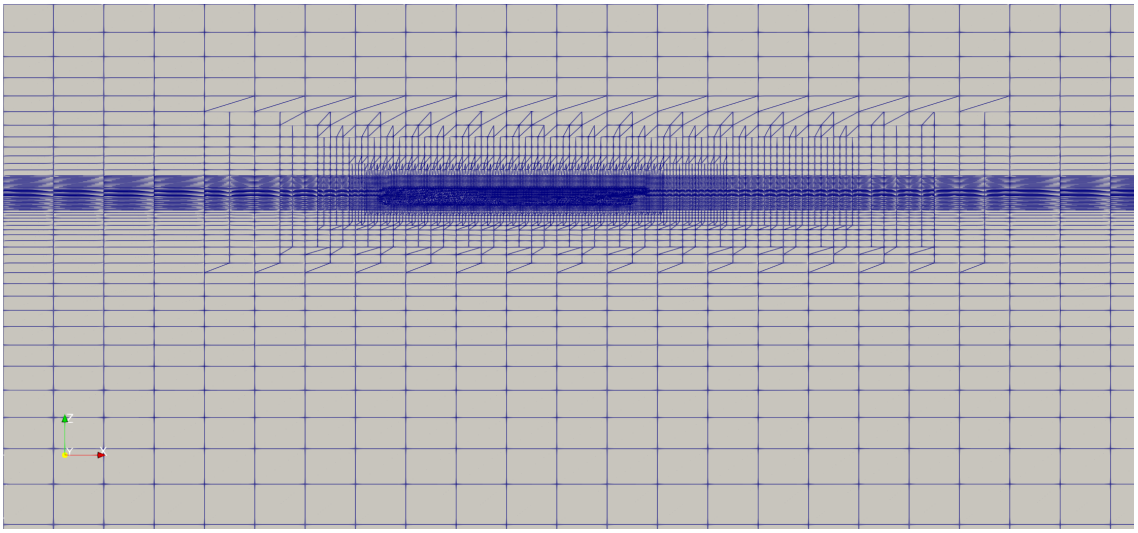
Since the main focus of this thesis is the development of an automated framework for trim optimization the requirements for the domain size, especially in the downstream direction are not as strict as the ones proposed by the IITC [25], to reduce the computational cost. At this point, it should be noted that due to the symmetrical nature of the examined problem in the xz plane, the half geometry will be solved for the reduction of the computational cost, without affecting the final value of the calm water resistance.

## 3.3 Mesh Generation

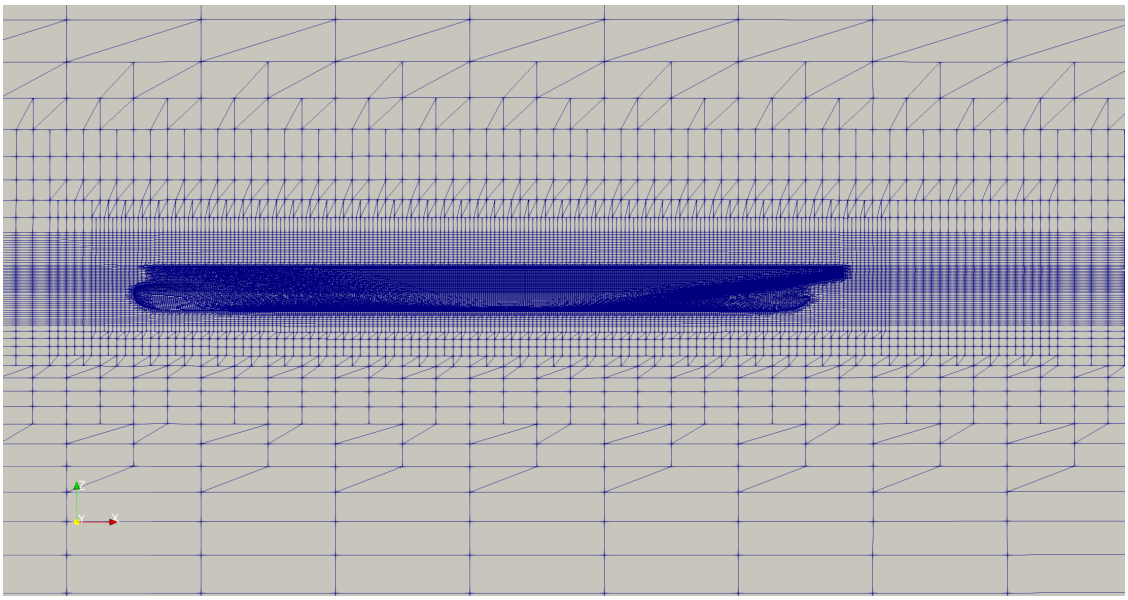
In the preceding section, the computational domain presented was not incorporated by the geometry. This is achieved by the snappyHexMesh utility provided with OpenFOAM. It is capable of creating 3-dimensional unstructured meshes composed of hexahedra and split-hexahedra over complex triangulated surface geometries given in Stereolithography (STL) format. The mesh generation process can be summarized in the following steps:

- The blockMesh option is utilized for the creation of the hexahedral cells consisitng the background mesh, in which the geometry is placed.
- Volume regions are generated in the domain, in different distances from the hull, that are then splitted with the command refineMesh for a denser domain.
- The edge features of the geometry are specified since they indicate the boundaries of the examined body. Then the cells are split into smaller hexahedrals based on the feature edges within the domain. The castellation continues to the neighbour cells of the surface.
- During this step the removal of the cells within the enclosed area of the surface takes place.
- The mesh created from the previous steps produced a jagged surface, which is smoothened by the snapping. The mesher adjusts the cell vertex points, of the already generated mesh, onto the STL surface, through an iterative process.

- The last stage of mesh generation is the creation of layers, which essentially creates hexahedral cells in line with the boundary surface, to replace the irregular cells created by the snapping process and to capture the boundary layer with greater detail.



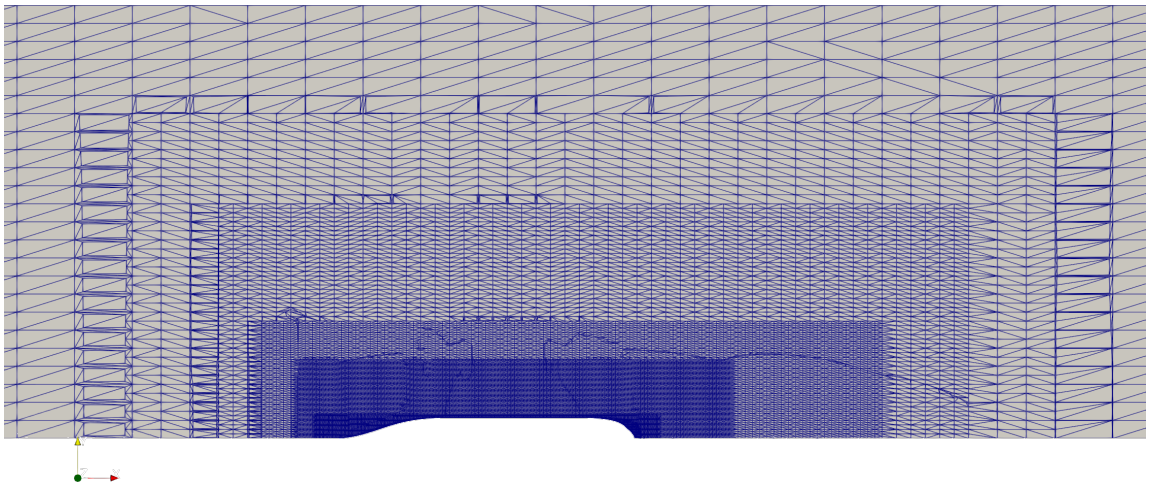
**(a)** *Distant perspective*



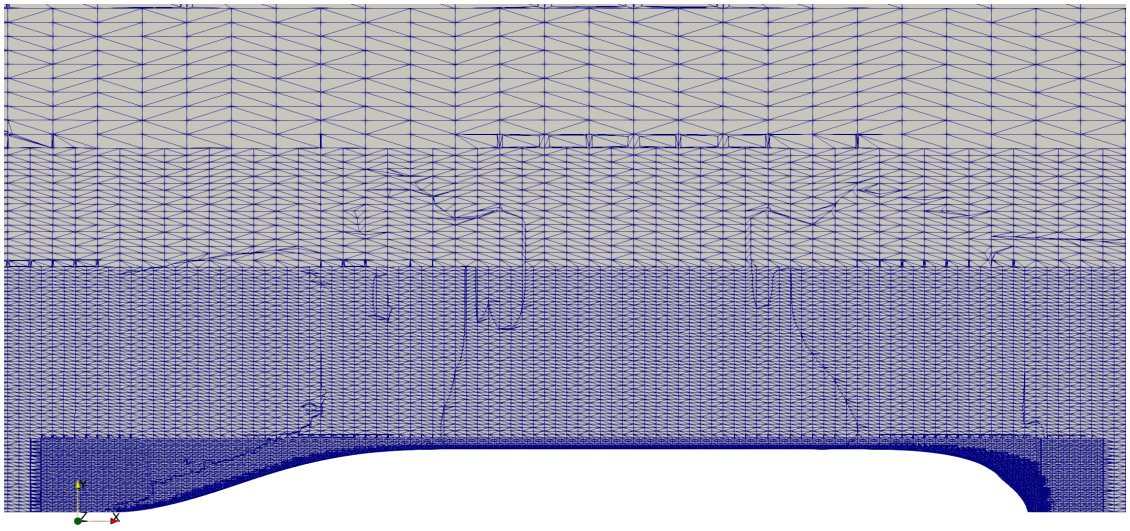
**(b)** *Close up perspective*

**Figure 3.3:** *Side view of the computational domain along with the mesh grid.*

As shown in figures 3.3 and 3.4, in the grid there are distinct boxes each one with a different refinement level, increased by one as the boxes get closer to the hull. This approach was adopted because of its versatility in examining different cell numbers while focusing on the main areas of interest. In the simulation of calm water resistance, it is crucial to capture the generated waves, known as Kelvin waves with a propagation angle equal to 19.5°; hence, the area astern of the vessel



**(a)** *Distant perspective*



**(b)** *Close up perspective*

**Figure 3.4:** *Top view of the computational domain along with the mesh grid.*

is denser in both the x and y direction. However, closer to the outlet the cells are coarser to reduce the creation of unnecessary cells and to assist with the damping of the generated waves.

A precise depiction of the pressure and velocity field would require approximately 60 to 80 grid points per wavelength. The wavelength of the longest wave formed behind a vessel is given by $\lambda = 2\pi Fn^2 L$ [16] and in the case of the KCS hull it is equal to $\lambda = 3.03m$. Taking into account the refinement regions, in the examined case the corresponding grid points are around 65. Also the requirement for 10 to 15 grid points in the height of the wave is satisfied.

## 3.4 Boundary Conditions

In the field of computational fluid mechanics the governing equations, and subsequently the linear system of the preceding Chapter 2 can describe a wide range of applications. Therefore, the proper boundary conditions, which correspond to the calculation of calm water resistance, must be specified. These conditions are applied at the patches, which are the regions designating the domain boundaries for the velocity, pressure, k, $\omega$, $\mu_t$ and $\alpha$.

The different boundary condtion types for each patch, as defined in OpenFOAM are:

- Inlet:
  - The inlet velocity has a fixedValue, essentially a Dirichlet condition, of magnitude equal to $(2.196, 0, 0)$.
  - The pressure condition is "fixedFluxPressure", which sets the pressure gradient to the value such that the flux on the boundary is specified by the velocity boundary condition.
  - For k a "fixedValue" is given equal to 0.00015.
  - As on k, *omega* has also a "fixedValue" equal to 2.
  - The eddy viscosity $\mu_t$ is "fixedValue" and equal to $5 \cdot 10^{-7}$.
  - Since the calm water resistance is of interest, the boundary condition for $\alpha$ is "fixedValue" and uniformly zero.
- Outlet:
  - For the velocity, the "outletPhaseMeanVelocity" condition is chosen. The velocity is adjusted for the given phase to get the given mean value, ensuring the same water level for both the inlet and outlet.
  - The pressure is a Neumann condition, which in the examined case is defined as "zeroGradient".
  - For k, the "inletOulet" condition is chosen, which is derived from Robin condition, and shifts between "zeroGradient" and "fixedValue" for the outflow and inflow, respectively.
  - As on k, *omega* has also an "inltetOutlet" condition.
  - The eddy viscosity $\mu_t$ boundary condition is "zeroGradient".

- The $\alpha$ is defined as "variableHeightFlowRate"; the volume fraction depends on the local flow conditions. For $\alpha > 1.0$ a "fixedValue" is applied, for $0.0 \leq \alpha \leq 1.0$ "zeroGradient" condition is applied and for $\alpha < 0.0$ "fixedValue" is applied. For both the first and latter case the value is equal to 0.

- Atmosphere:
  - The inlet velocity is "pressureInletOutletVelocity". Similar to the "inletOutlet", for the outflow the "zeroGradient" condition is applied while for the inflow the velocity is based on the flux in the patch-normal direction.
  - The pressure condition is "prghEntrainmentPressure" and is calculated as follows: $p_{rgh} = p - \rho g h$, where $h$ is the height, $p = p_0 + 0.5\rho U|U|$ is the static pressure and $p_0$ is the total pressure and is equal to zero in all directions.
  - The k is defined as "inletOutlet".
  - As on k, *omega* is also "inletOutlet".
  - The eddy viscosity $\mu_t$ is "zeroGradient".
  - The VOF $\alpha$ is "inletOutlet".

- Hull:
  - The inlet velocity is "movingWallVelocity" and sets the value of the velocity magnitude equal to $(0, 0, 0)$, since the no-slip condition must be satisfied.
  - The pressure condition is "fixedFluxPressure".
  - For k a "kqRWallFunction" is given.
  - The *omega* is "omegaWallFunction".
  - The eddy viscosity $\mu_t$ is "nutkRoughWallFunction" and provides a wall constraint on the turbulent viscosity, based on the turbulent kinetic energy.

The rest patches are treated as symmetry planes with the corresponding boundary condition.

## 3.5 Numerical Schemes and Solution Parameters

In the preceding sections, the method for the mesh generation, as well as the required boundary conditions for the specific problem have been described. However, the discretization of the terms in the equations presented in Section 2.2, are yet to be determined. In the OpenFOAM environment, the option for specifying the numerical integration schemes, for the surface and volume integrals, is provided. Also, corrections and limiters can be implemented on top of the schemes for better stability of the solution. Since, the cell-centred approach is implemented, for the storage of the unknown variables, an interpolation scheme is then applied for calculating those values from the cell centre to the cell face.

Due to the complexities and transient phenomena being present in the calculation of the ship's calm water resistance, an unsteady approach is required. Hence, the Euler implicit time scheme is imposed; the accuracy provided by this scheme is adequate in the context of this study despite being a first order scheme. As already mentioned, OpenFOAM utilizes the cell-centred Finite Volume framework, therefore to extract the result on the cell faces a linear interpolation was selected. A linear scheme was selected for both the laplacian and gradient schemes. However, for the latter limitations were applied for the cell gradient to ensure, that the extrapolated values on the cell face are constrained by the minimum and maximum values of the surrounding cells. Finally, a second-order linear upwind scheme was elected for the discretization of the convection terms, in order to achieve better numerical stability and a greater representation of the upwind flow behavior. The unstructured grid constructed for the numerical solution can introduce errors and inaccuracies in the simulation result; thus non-orthogonal correction method was introduced to mitigate the effects caused by the misalignment of the grid lines.

**Table 3.3:** *The discretization and interpolation schemes as defined in the OpenFOAM environment.*

| Symbol | Name | Scheme |
|---|---|---|
| t | ddtSchemes | Euler |
| grad | gradSchemes | cellLimited Gauss linear 1 |
| lap | laplacianSchemes | Gauss linear corrected |
| int | interpolationSchemes | linear |
| div | div(rhoPhi,U) | Gauss linearUpwind grad(U) |
| div | div(phi,alpha) | Gauss interfaceCompression vanLeer 1 |
| div | div(phirb,alpha) | Gauss linear |
| div | div(phi,k) | Gauss linearUpwind limitedGrad |
| div | div(phi,omega) | Gauss linearUpwind limitedGrad |

The PISO algorithm, as explained in section 2.5.1, is then initiated to numerically

solve the linear system of the governing equations. The process is dictated by user defined settings. Firstly, the simulation time was set to 25 seconds (simulation time) with a time step equal to 0.002 seconds to ensure a low value for the Courant number (maximum value is set to 1) for a better stability and to capture the wave creation with detail. Also, the internal iterations within a time step are controlled by the specified residual, which for the examined case is equal to 1e-08 and 1e-07 for the volume of fraction and for the rest variables, respectively.

Finally, the simulations were performed in a computer provided by the Laboratory of Ship and Marine Hydrodynamics Laboratory of Ship and Marine Hydrodynamics NTUA equipped with an Intel Core i7 CPU with 4 cores and 2.80 GHz clock speed and 16 Gb RAM. The computational domain was decomposed, and the case run in parallel utilizing all the available cores, and the required physical time was about 46 h per case for the medium-sized grid as described in Section 3.7.

## 3.6 Forces Calculation

The final step in the numerical approach of the calm water resistance is the calculation of the forces acting upon the body. The forces are generally divided into two categories, one due to the waves and the pressure gradients and one caused by the viscous resistance of the submerged geometry, called pressure and viscous force, denoted with the symbols "p" and "v" respectively in the OpenFOAM environment and calculated as follows.

$$F_p = \int_S p\mathbf{n}dS \tag{3.1}$$

$$M_p = \int_S (\mathbf{x}_f - \mathbf{x}_0) \times p\mathbf{n}dS \tag{3.2}$$

where p is pressure, n is the vector normal to the cell face and $\mathbf{x}_f$ is the x-coordinate of the body's centre of rotation.

$$F_\nu = \int_S \tau_{ij}\mathbf{n}dS \tag{3.3}$$

$$M_\nu = \int_S (\mathbf{x}_f - \mathbf{x}_0) \times \tau_{ij}\mathbf{n}dS \tag{3.4}$$

where $\tau_{ij}$ is the stress tensor.

The CFD calculation provides the forces for each direction, but since the hull's resistance is only of interest, the forces in the x-direction are taken into account. Finally, the total resistance coefficient for calm water is [16]:

$$C_T = \frac{R_T}{\frac{1}{2}\rho V^2 S} \tag{3.5}$$

where, $R_T = F_{px} + F_{\nu x}$ is the total calm water resistance and S is the wetted surface of the KCS geometry as defined at table 3.1.

## 3.7 Mesh Sensitivity Analysis

Before proceeding with the results of the calm water resistance and subsequently with the trim optimization a mesh sensitivity analysis was conducted, to determine the effect of the number of cells on the coefficient of the total resistance. This step is crucial in the context of this thesis since it could save from excessive computational time. Three different scenarios for the number of cells were examined each of which had a denser grid compared to the previous one. The final values of the total resistance were then compared with the available experimental data [8] by calculating the deviation for each scenario, as follows.

$$E\% = \frac{D - S}{D} \times 100\% \tag{3.6}$$

where, D and S are the experimental (EFD) and CFD results respectively.

**Table 3.4:** *Grid size and results for mesh sensitivity.*

| **Grid** | **Number of cells** | $\mathbf{C_T \times 10^3}$ | | **Deviation [%]** |
|---|---|---|---|---|
| | | **CFD** | **EFD** | |
| Fine | 1675043 | 3.63 | 3.71 | 1.92 |
| Mid | 1206743 | 3.67 | | 0.92 |
| Coarse | 662152 | 3.77 | | -1.82 |

From Table 3.4, the results showed an expected behaviour and were considered acceptable, especially in the context of this study, where the hull motion was restricted adding significant deviations to the resistance value. On the other hand, The results of the fine mesh were not as close to the EFD data as supposed, despite having a higher number of cells, because the refinement in the key areas of the hull was not adequate to capture the flow. However, the investigation of an even denser mesh could not be conducted due to the limitations opposed by the available hardware, making the required computational cost unviable. Additionally, it was concluded

that the best grid option would be the mid mesh because it provides a good approximation of the resistance value while having a relatively small number of cells. These advantages are crucial in trim optimization, where multiple simulations were conducted, and the mesh was regenerated for each condition. In the mesh sensitivity analysis, all the solution parameters, as described in Section 3.5 were the same among the different grid sizes to see only the effect from the mesh resolution.

Through the previous sensitivity analysis, a baseline grid resolution has been established for the calm water resistance simulation, so several speeds were examined to assess the behaviour of the grid in different conditions with varying waves. In Figure 3.5, the total resistance coefficient is plotted for various Froude numbers to obtain the resistance curve of the model. It is apparent that a good correlation with the experimental data has been achieved, with the exemption of the lower speed. As already explained, the Fn number is an indication of the wave resistance therefore, for low Fn values, the generated waves are weak, and the mid-grid cannot capture with detail the characteristics of the wave system. However, for the procedure of trim optimization, which will be investigated in the following chapter, the Froude number equal to 0.26 was chosen.

**Table 3.5:** *Total Resistance at varying Froude numbers.*

| **Fn** | **Resistance [N]** | **$C_T \times 10^3$** | | **Deviation [%]** |
|---|---|---|---|---|
| | | **CFD** | **EFD** | |
| 0.195 | 23.77 | 3.677 | 3.475 | -5.81 |
| 0.227 | 29.97 | 3.421 | 3.467 | 1.31 |
| 0.260 | 42.26 | 3.677 | 3.711 | 0.92 |
| 0.282 | 60.77 | 4.494 | 4.501 | 0.15 |

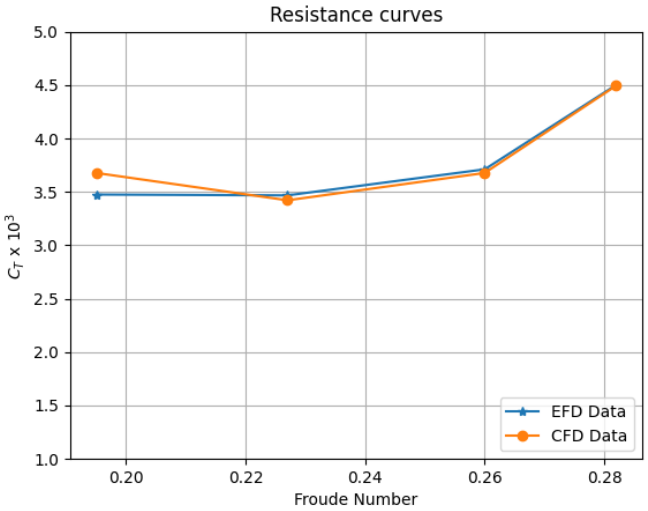



**Figure 3.5:** *EFD and CFD data of the total resistance coefficient for the KCS hull at varying speeds.*

## 3.8 Calm Water Resistance-Results

In this section, the results of the calm water resistance will be presented, as a reference for comparison with the results obtained by the trim optimization. The simulation setup was done according to the analysis presented in the preceding sections and the KCS bare hull is at evel keel with a Froude number equal to 0.26.

In Figure 3.6, the convergence of the viscous and pressure force over simulation time is plotted. It is evident that the viscous force converged quickly compared to the pressure force, which shows an oscillating behaviour, because of the time needed for the wave creation. This behaviour was expected because there was not a damping zone at the outlet of the numerical towing tank and the shape of the stern; however, the magnitude of the oscillation is small and as a result, the influence on the bare hull resistance is not significant, especially in the context of this thesis where the goal was the development of a trim optimization framework and not to predict as accurately as possible the total resistance. Also, for this purpose, it was decided to reduce the simulation time to 15 s to reduce the total time of the optimization.

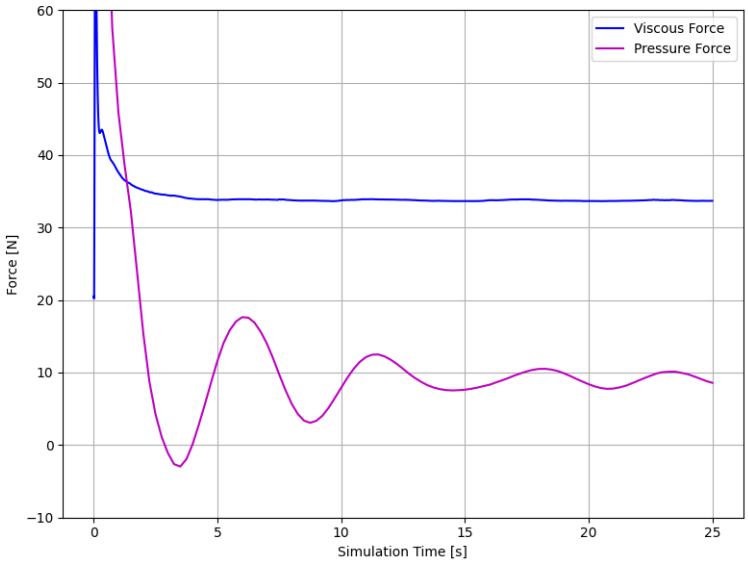



**Figure 3.6:** *Viscous and Pressure Force convergence for Froude number 0.26*

Figure 3.7, represents the free surface (VOF equal to 0.5), of the problem where the wave pattern is illustrated by plotting the position in z-direction, followed by the pressure distribution along the hull.

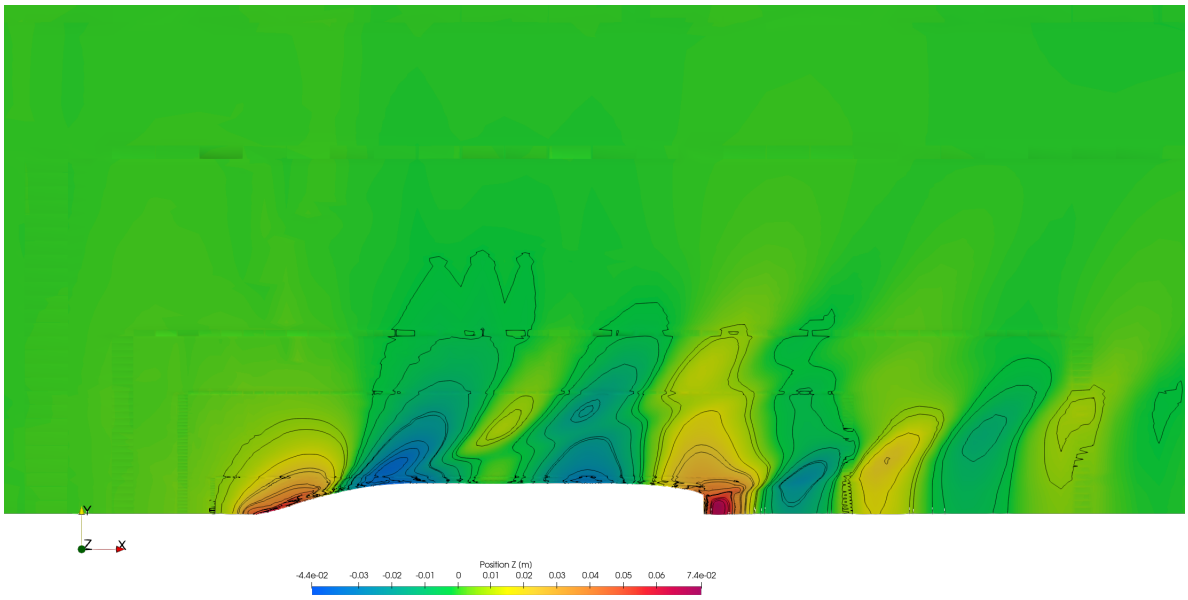
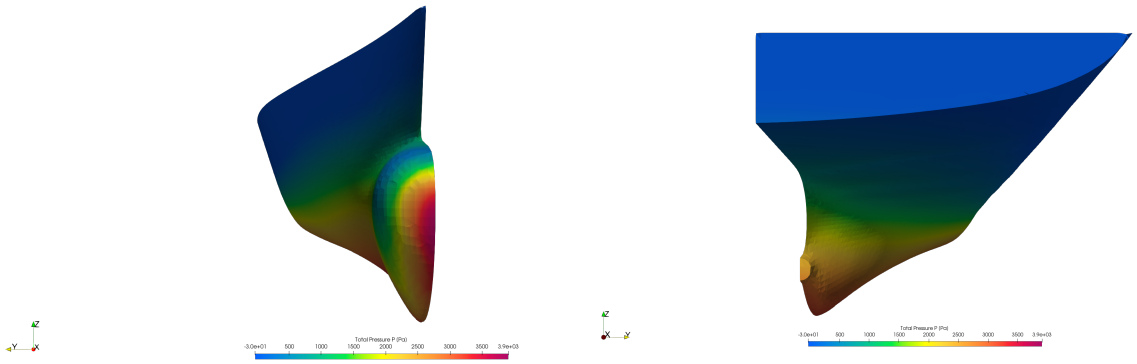
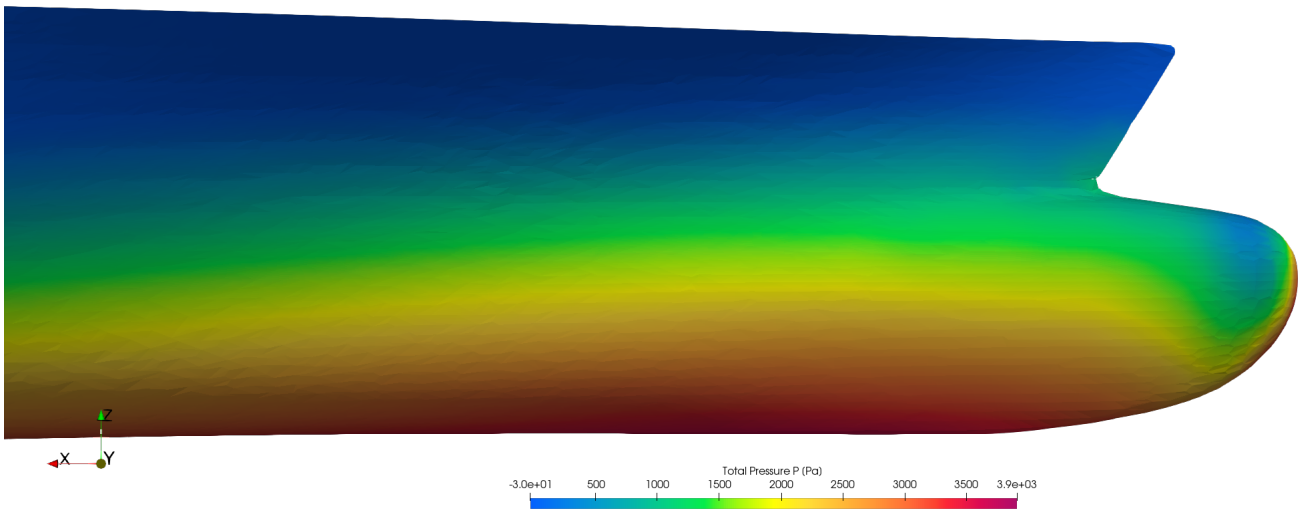



**Figure 3.7:** *Height of the Kelvin waves on the free surface at even keel for Froude number 0.26*



**(a)** *Stem View*

**(b)** *Stern View*




**(c)** *Side view of the bulbous area.*

**Figure 3.8:** *Pressure distribution on the KCS hull at even keel for Froude number 0.26, in the bulbous area*

# Chapter 4

# Trim Optimization

## 4.1 Introduction

The goal of this thesis is to optimize the ship's performance by minimizing the resistance. This problem continuously concerns engineers, as the pursuit of a better design solution never ends. In this chapter, a workflow for optimizing the ship's trim and sinkage is developed. The objective function of the optimization is the resistance, which was calculated in the previous chapter. A brief description of the COBYLA algorithm used for finding the optimum value is given before presenting the steps that were followed for establishing an automated framework for the trim optimization and subsequently for the trim and sinkage optimization. Finally, an illustration of the CFD results for each scenario is given.

## 4.2 Optimizer

Optimization is the process of defining the optimal solution for a given problem. This in mathematical modelling can be expressed as the task of finding the minimun or maximum value of one or more objective functions F subject to constrains, which in simple cases can be achieved by finding the values for which the gradient of the base function is zero. However, there are cases in which no constrains are applied. The objective function in the problem of trim optimization is the total resistance of the vessel and it falls into the category of constrained problems, as the trim angle is dependent on many parameters, mainly by the stability. Because, the resistance cannot be determined by a formula, as the pressure field cannot be solved analytically, an alternative approach was implemented.

The problems where the derivative of the objective function is unavailable or hard to compute fall into the category of Derivative-free optimization. However, for the in-

vestigated problem no information was present, even for approximating the gradient via finite differences; thus, the derivative-free COBYLA (Constrained Optimization BY Linear Approximation) algorithm was adopted. COBYLA is capable of solving non-linear optimization problems subject to constraints.

COBYLA iteratively defines a simplex on which the objective function is linearised, by linear interpolation between the values at the vertices, and optimised. Afterwards, a new simplex is created either by expanding, contracting or reflecting the original one. The criteria for choosing the new simplex is to improve the value of the objective function or the simplex shape. The initial size of the simplex's edges is equal to $\rho_{start}$, and is gradually reduced during the optimization process, until it becomes equal to $\rho_{end}$, when the process is terminated. In figure 4.1, a depiction of the two dimensional COBYLA algorithm is shown. Finally, through the implementation of a penalty function, which is adapted in the iteration process, the constrains are enforced.

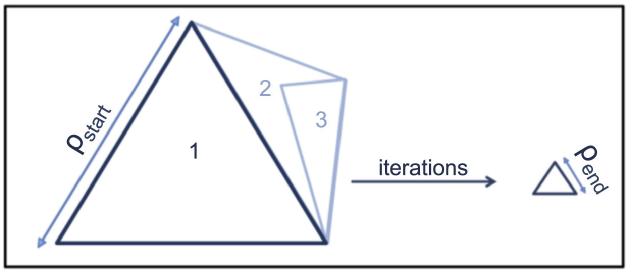



**Figure 4.1:** *COBYLA algorithm in two dimensions. The initial and final simplexes with edge size $\rho_{start}$ and $\rho_{end}$, respectively are illustrated.*

## 4.3 Trim optimization procedure

The trim optimization idea in this thesis was to couple two individual programs and for that reason, a connecting bridge should be established. This was achieved by creating a script written in Python language, as the OpenMDAO, which can extract the data from the numerical simulation and pass them to the optimizer as input for finding the optimum trim angle.

Furthermore, OpenFOAM is a powerful tool for CFD applications because it is designed to be driven by the command line; hence, it provides an ideal environment for automated procedures and batch processing. On the other hand, OpenMDAO is a Python module explicitly utilized through the execution of the corresponding programming language. On that basis, the necessity for creating a script that acts as a communication channel emerged; for that purpose, the programming language Python and the command language Bash were employed. Python was selected for

its simplicity, versatility and the wide eco-system of third-party libraries, which include modules for various applications reducing the need to write code from scratch. Bash, however, despite having a steep learning curve, performs exceptionally for the OpenFOAM applications, as the integration of the simulation into an executable script can automate the whole process without the need for additional software, as it is available by default on Unix systems.

The first step in the development of an automated framework was the creation of an executable script written in Bash to perform the simulation of the calm water resistance. By doing so, the simulation made the need for the user's input obsolete, and the simulation could be initiated by everyone having access to the command prompt. Afterwards, the OpenMDAO, was utilized for creating a code to assemble the optimization model by defining the design variables (input for the optimization algorithm), which in the examined case is the resistance, constraints and the preferred optimizer, who was responsible for providing the trim angle (output). The problem that arose next was how to use the output of the simulation as an input to the optimizer and then use the output from the optimizer for rotating the geometry. The solution was given by writing another script but in Python this time that was responsible for reading the output of the optimizer using this value for transforming the geometry and then initiating OpenFOAM with the updated geometry.

The procedure described above is summarized in Figure 4.2 and consists of the following steps:

1. The process starts by calling the code that creates the optimization model.
2. Obtain the trim angle from the optimizer.
3. The geometry is rotated based on the updated trim angle.
4. The mesh grid is generated for the new geometry.
5. Check if the desired mesh quality is achieved; otherwise, go back to the previous step and re-run the mesher.
6. Initiate the solver to get the calm water resistance.
7. Extract the value of the total resistance.
8. Use the value from the previous step, as input for the optimizer.
9. If the problem has achieved convergence, the process is terminated; otherwise, the trim angle is updated and goes back to step 2.

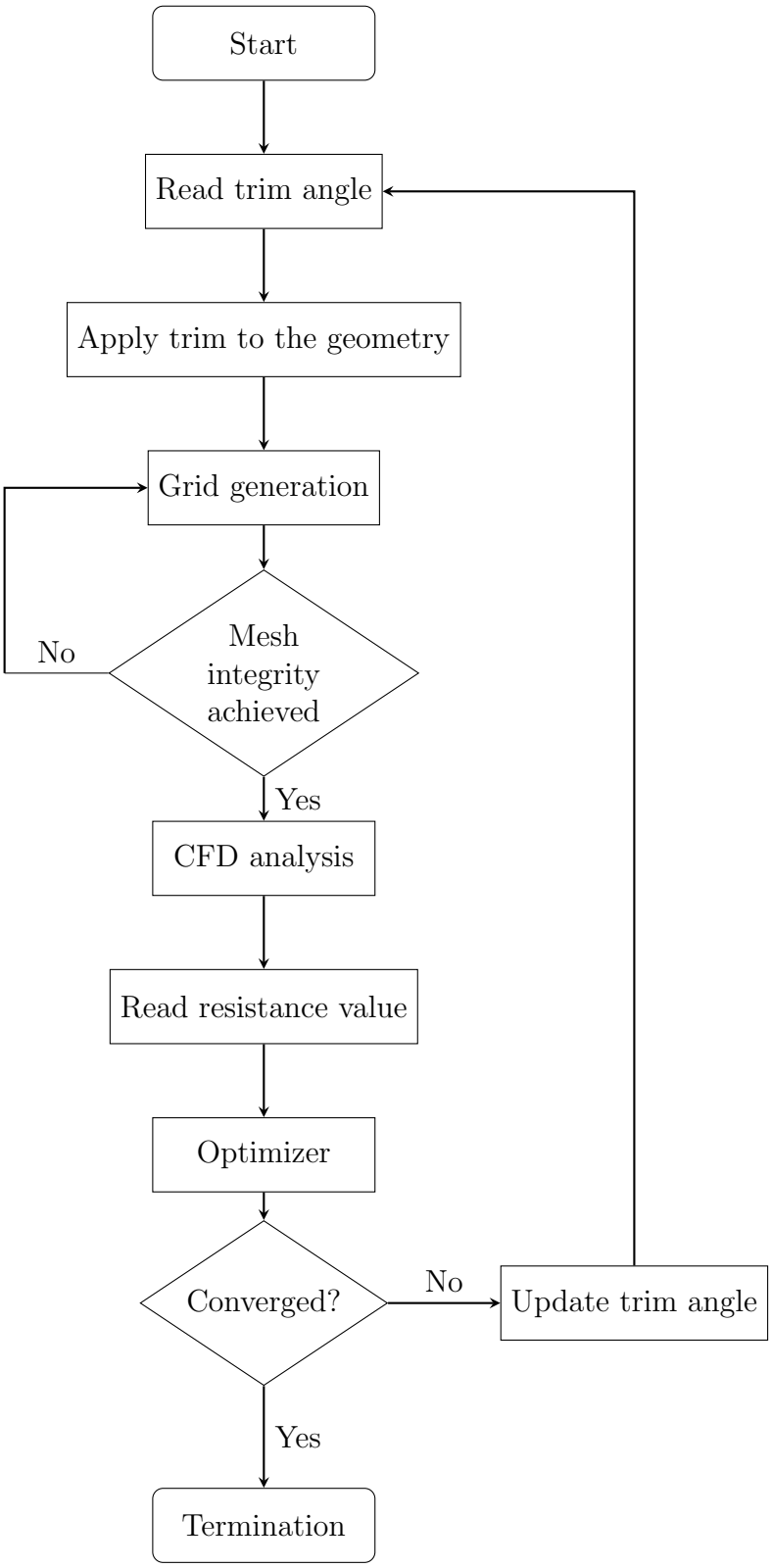



**Figure 4.2:** *A flow chart of the trim optimization process*

## 4.4 Results

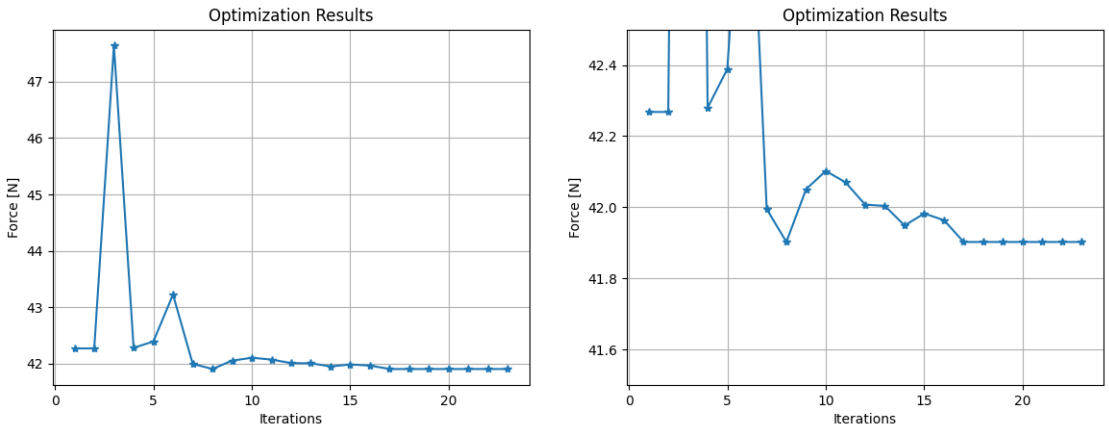
### 4.4.1 Trim angle

The COBYLA algorithm upon finilization concluded that trim angle has a significant effect on the total resistance, as expected and the optimum value was $-0.250°$; the notation followed in this study indicates that trim by the bow is negative while trim by stern is positive. The results, as presented at table 4.1, indicate a 0.85% decrease in the value of total resistance.

**Table 4.1:** *Total resistance for the initial and optimum trim condition.*

| Conditon | Trim angle [°] | Resistance [N] | Difference [%] |
|---|---|---|---|
| Initial | 0.0 | 42.26 | -0.85 |
| Optimum | -0.250 | 41.90 | |

In Figure 4.3, the history of the trim optimization cycle is presented, showing a good convergence of the COBYLA algorithm. However, the last few iterations saw no improvement in the resistance value due to the strict convergence criteria enforced; thus, a reduction in the overall needed optimization time can be achieved by examining further the best termination criteria. Additionally, the strange behaviour and the extreme results of the first few steps are attributed to the optimization algorithm, which is trying to establish the trust region. Sometimes, COBYLA might also ignore the constraints, but the final value of the optimization always lies within the boundaries specified by the user.




**(a)** *The full results history.* **(b)** *Zoomed view of the results history.*

**Figure 4.3:** *The value of total resistance at each trim optimization iteration.*

To get a better understanding of the optimum trim condition, the CFD results of the calm water resistance were visualized, as shown in Figures 4.4 and 5.3. The generated waves had a slighter larger amplitude, in comparison with the ones at

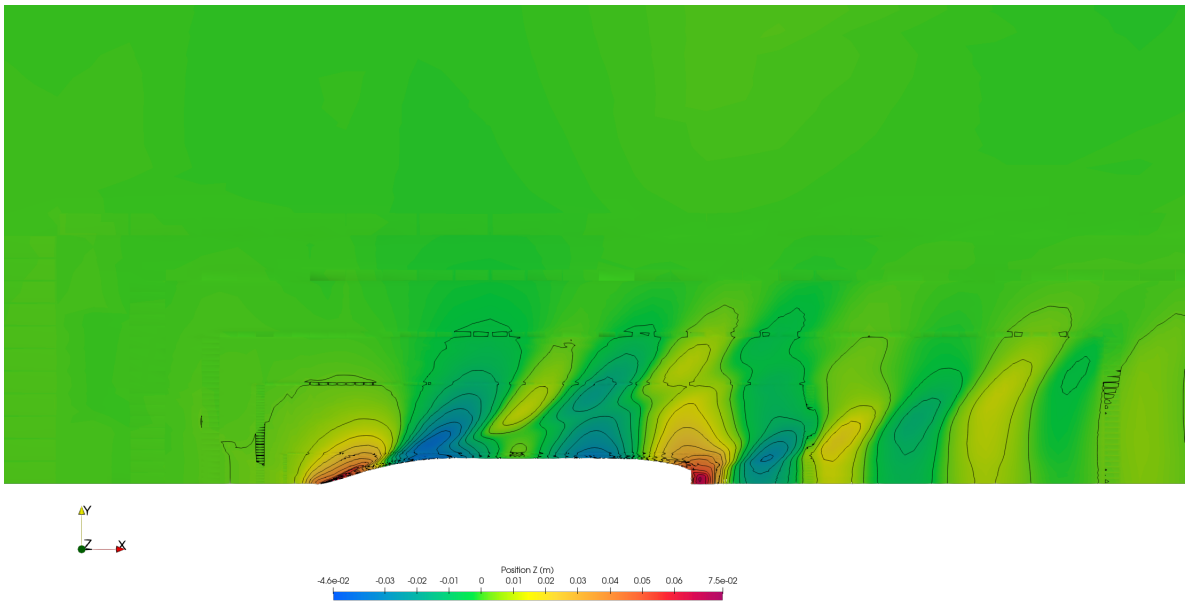
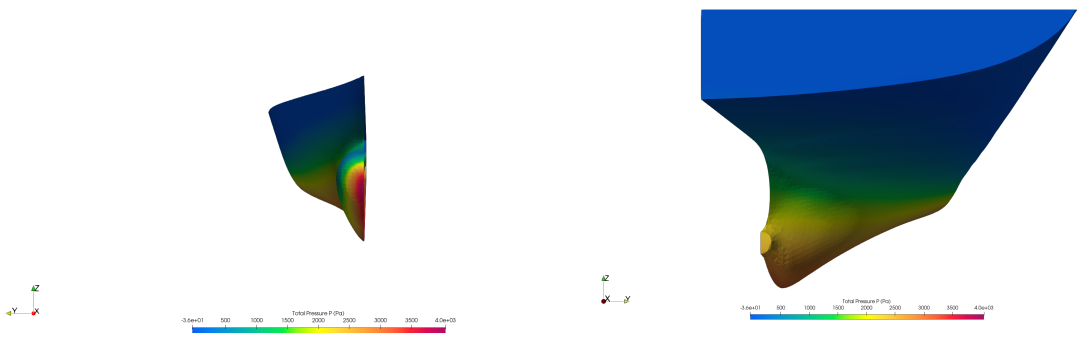


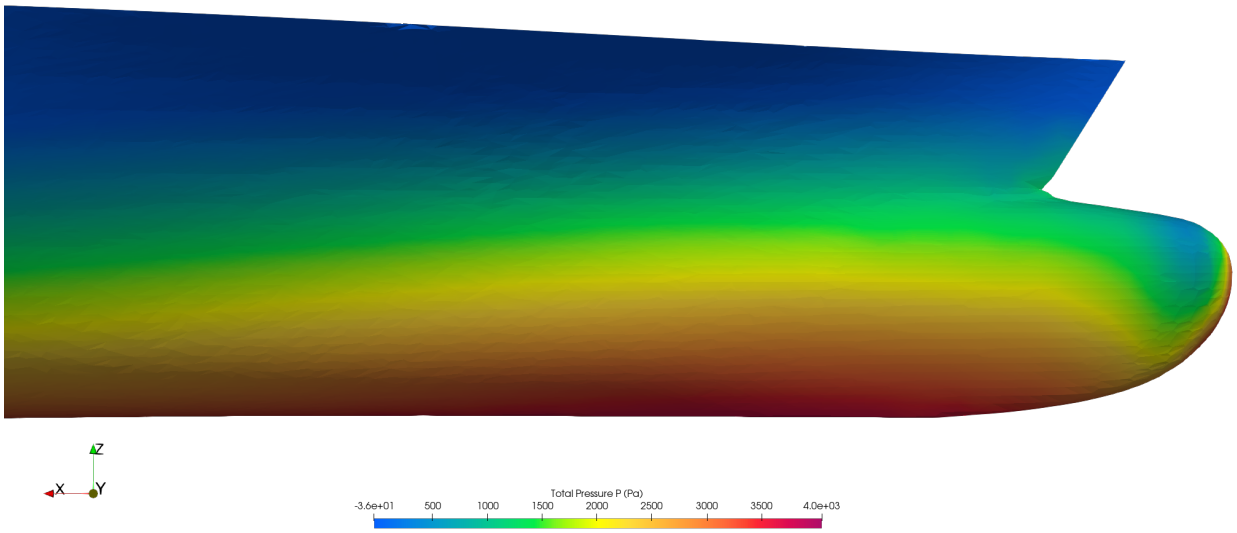

**Figure 4.4:** *Height of the Kelvin waves on the free surface at optimum trim condition for Froude number 0.26.*

even keel condition, Figure 3.7, but the spread in the y-direction is not as wide. Additionally, the separation area in the wake of the ship, caused by the form of the stern, was concentrated closer to the hull with less spread, and the waves dissipated quicker.



**(a)** *Stem View*

**(b)** *Stern View*




**(c)** *Side view of the bulbous area.*

**Figure 4.5:** *Pressure distribution on the KCS hull at optimum trim conditon for Froude number 0.26, in the bulbous area*

### 4.4.2 Trim angle and sinkage

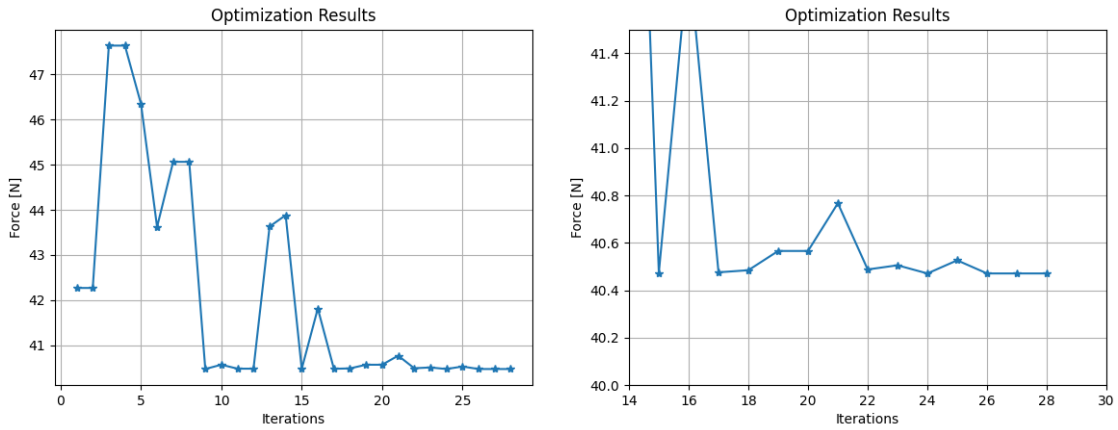
The concept of trim optimization was extended further by investigating the combination of the trim angle and draught. This approach, was selected to exhibit the applicability of the developed framework for a case with more than one design variable; in most cases, the draught reduction equals a decrease in the total resistance because the wetted surface is significantly reduced. For vessels with bulbous, the increase of draught from a point and onwards will have undesired effects because the bulbous, when partially submerged does not decrease the wave-making resistance as designed; hence, the total resistance increases despite the reduction in the friction resistance. Also, for the majority of the journey ships operate in the maximum draught allowed, leaving no room for draught adjustment. To prevent an extreme decrease of the draught the variable representing the sinkage was constrained.

**Table 4.2:** *Total resistance for the initial and optimum trim and sinkage condition.*

| **Conditon** | **Trim angle** [°] | **dT** [m] | **Resistance** [N] | **Difference** [%] |
|---|---|---|---|---|
| Initial | 0.000 | 0.000 | 42.26 | -4.23 |
| Optimum | -0.250 | -0.015 | 40.47 | |

As in the preceding section, the iterations of the optimization cycle were plotted, as shown in Figure 4.6, and the first few repetitions exhibit similar behaviour to

the one at the trim optimization, Figure 4.3. This time the iterations needed for the convergence were increased due to the complexity introduced to the problem by adding one extra design variable, but this time the convergence criteria were not as strict and as a result, fewer iterations, in the end, concluded the same value for the resistance.




**(a)** *The full results history.* **(b)** *Zoomed view of the results history.*

**Figure 4.6:** *The value of total resistance at each trim and sinkage optimization iteration.*

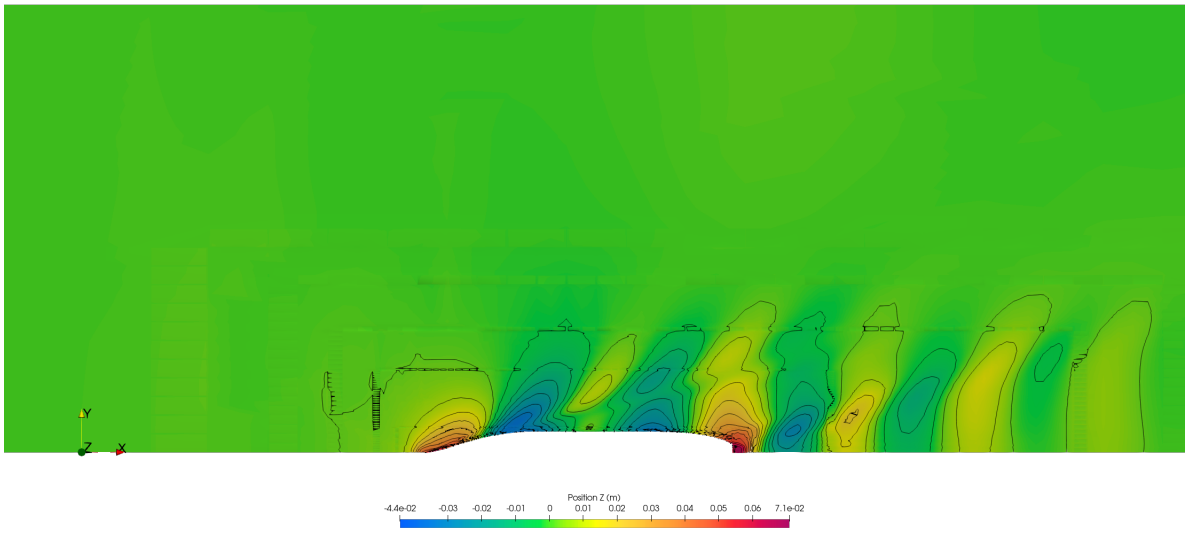
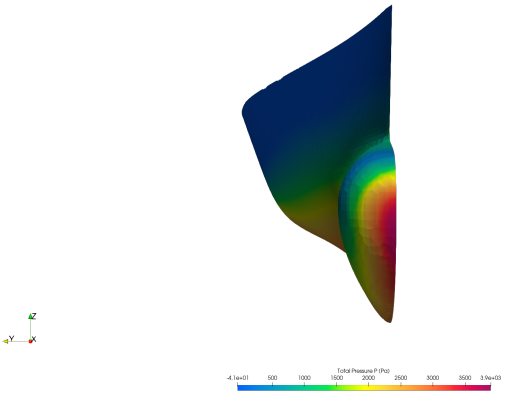
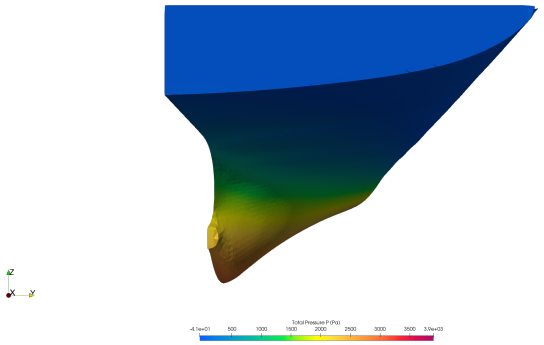


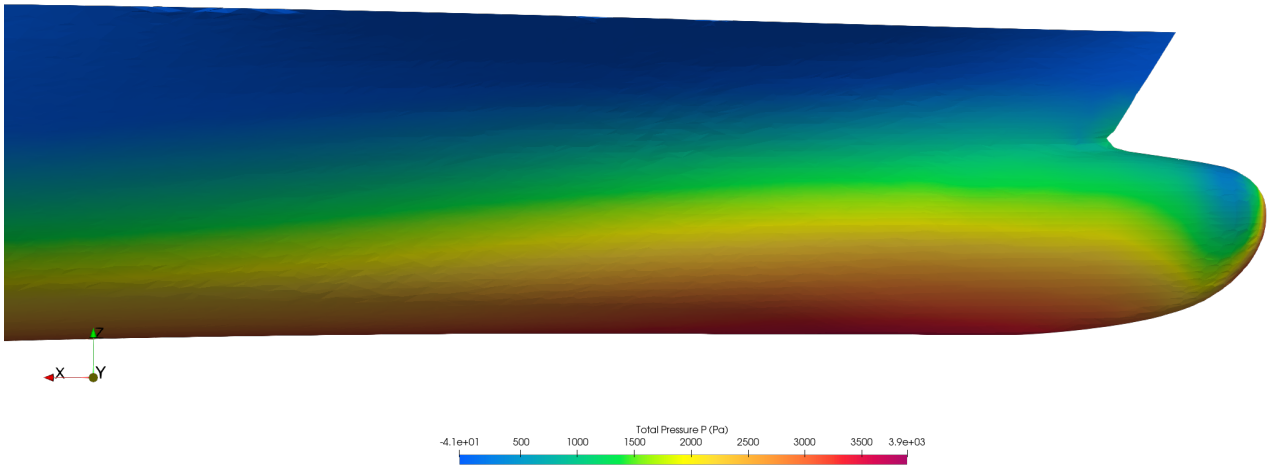

**Figure 4.7:** *Height of the Kelvin waves on the free surface at optimum trim and sinkage condition for Froude number 0.26.*



(a) *Stem View*



(b) *Stern View*




(c) *Side view of the bulbous area.*

**Figure 4.8:** *Pressure distribution on the KCS hull at optimum trim and sinkage conditon for Froude number 0.26, at the bulbous area*

# Chapter 5

# Bulbous Shape Optimization

## 5.1 Shape parametrization

As already demonstrated in the preceding chapters, the wave resistance and subsequently, the trim optimization are influenced by the bulbous shape. Hence, in this section, bulbous form optimization is being investigated. First, the theoretical background for the parametrization and the deformation of a given object is presented, followed by the implementation of the method in the KCS bare hull geometry. The chapter is concluded with the results obtained from the hull form optimization.

### 5.1.1 B-Spline curves

A B-spline curve $C(u)$, is defined as follows:

$$C(u) = \sum_{i=0}^{N_u-1} N_{i,p_u}(u) P_i \tag{5.1}$$

where, $P_i$, $i = 0, 1, ..., n$ are the control points and $N_{i,p_u}(u)$ is the i-th basis function, with degree $p_u$. The curve given from equation above is a piecewise polynomial function, with a maximum degree of $p_u$ for each polynomial.

For the definition of the basis function $N_{i,p_u}(u)$, a set of knots in ascending order, called knot vector $\mathbf{T}$ containing $n + p_u + 1$ elements, should be defined.

$$\mathbf{T} = \left[ \underbrace{0, \cdots, 0}_{p_u+1}, \frac{1}{N}, \cdots, \frac{N+1}{N}, \underbrace{1, \cdots, 1}_{p_u+1} \right] \tag{5.2}$$

where, $N = n - p_u + 1$. The format of the knot vector in equation 5.2, results

to curves known as clamped because the first and last control points of the curve coincide with the endpoints of the curves.

The zero-order basis function is defined by:

$$N_{i,0}(u) = \begin{cases} 1, & t_i \leq u \leq t_{i+1} \\ 0, & \text{elsewhere} \end{cases} \tag{5.3}$$

The higher-order basis functions are given by the recursive formula as follows:

$$N_{i,p}(u) = \frac{u - t_i}{t_{i+p} - t_i} N_{i,p-1}(u) + \frac{t_{i+p+1} - u}{t_{i+p+1} - t_{i+1}} N_{i+1,p-1}(u) \tag{5.4}$$

Equation 5.1 can be further extended, for two and three dimensional curves, by adding extra control points in the y and z directions.

For surfaces we have:

$$S(u, v) = \sum_{i=0}^{N_u - 1} \sum_{j=0}^{N_v - 1} N_{i,p_u}(u) N_{j,p_v}(v) P_{i,j} \tag{5.5}$$

For volumes we have:

$$V(u, v, w) = \sum_{i=0}^{N_u - 1} \sum_{j=0}^{N_v - 1} \sum_{k=0}^{N_w - 1} N_{i,p_u}(u) N_{j,p_v}(v) N_{j,p_w}(w) P_{i,j,k} \tag{5.6}$$

### 5.1.2 Free-form deformation

Free-form deformation (FFD) originates from the computer graphics field and was first described by [31], for deforming locally or globally solid geometric models. It utilizes the trivariate Bernstein polynomials approach, but since then the method has transformed to employ trivariate Bezier, B-spline or NURBS volumes.

For a better understanding of the functioning of FFD (Free-Form Deformation), one can envision placing an object or any geometrical form within a transparent, flexible, rubber-like structure. In essence, FFD enables the integration of the object's geometry into a manipulable volume by adjusting the points located on the surface, commonly referred to as FFD points.

The first step in the process of the FFD is to create the box, which is basically a rectangular prism, by defining the FFD points.

The design surface is then described in a mathematical formulation to apply the

deformation; by applying a least-squares regression to get a B-spline approximation of the volume, as defined in equation 5.6, in cartesian coordinates. However, a mapping for a given point, with coordinates $\mathbf{r} = [x_r, y_r, z_r]^T$ of the B-spline to the parametric space is necessary and calculated by solving the system of equations below.

$$\overline{R}(u,v,w) = \begin{bmatrix} x(u,v,w) - x_r = 0 \\ y(u,v,w) - y_r = 0 \\ z(u,v,w) - z_r = 0 \end{bmatrix} \tag{5.7}$$

The Jacobian $\frac{\partial V}{\partial u_i}$, $i\epsilon\,\{1,2,3\}$ is computed and inverted, prior to solving the system of equations by utilizing the Newton method. The Jacobian matrix is obtained by analytically calculating a closed-form expression derived from differentiating equation 5.6 with respect to the components of u.

Finally, the original geometry is replaced by a new B-spline volume with updated control points $P^{new}_{i,j,k}$, from which the deformation of the enclosed geometry derives.

In figure 5.1, an example of the FFD application is given; the first picture from left to right illustrates the original geometry embedded in the FFD box. Also, the following steps of mapping and remapping are presented before the final deformed geometry, denoted by a new B-spline volume and control points.

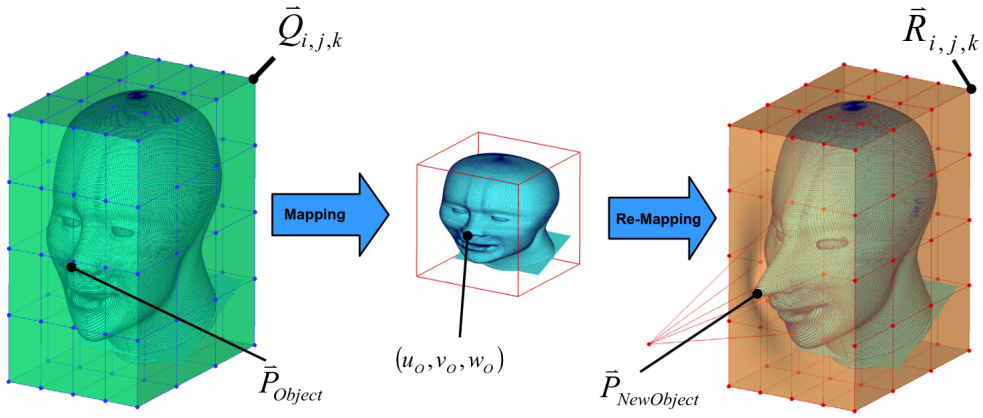



**Figure 5.1:** *Free-form deformation principle.*

The FFD procedure can be summarized in the following three steps:

- Creation of the FFD box with volumetric control points, enclosing the design surface.

- Indentify the mapping $\mathbb{R}^3(x, y, z) \rightarrow \mathbb{R}^3(u, v, w)$, from the physical coordinates to the parameter space, after embeding the design surface to the FFD box.
- Moving the FFD control points accordingly, to deform the original geometry.

## 5.2 Optimization procedure

Shape's form optimization objectives are the parametrization and deformation of the geometry; in the previous section, the theoretical background for tackling those objectives was given. Hence, the theory should be applied to the KCS hull to reshape the bulbous geometry, and for that purpose, the package pyGeo[1], as proposed by [14], was used; pyGeo is a geometry manipulation framework for multidisciplinary optimization developed by MDO Lab. It fits perfectly the requirements of this study since it is written in Python, as the other optimization scripts that have already been developed and because it utilizes the FFD approach. Furthermore, pyGeo parametrizes the changes of the geometry instead of the original geometry making the need for CAD software obsolete.

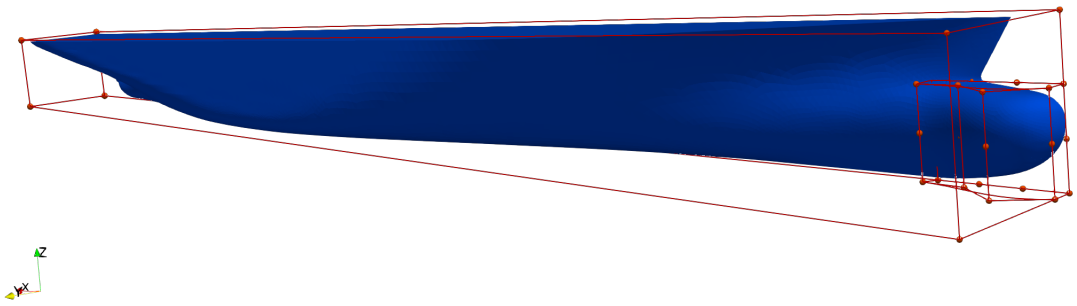



**Figure 5.2:** *The FFD box at the KCS geometry.*

The process of reshaping the bulbous consists of two steps; the parametrization of the geometry with the help of control points and then the geometry update by displacing the control points accordingly in each direction. The KCS hull so far was available in .stl format for the trim optimization, which was suitable for handling the geometry with pyGeo. A Python script was developed, for the parametrization of the geometry with the help of B-spline volumes. The code's input was the STL file,

[1]https://github.com/mdolab/pygeo.git

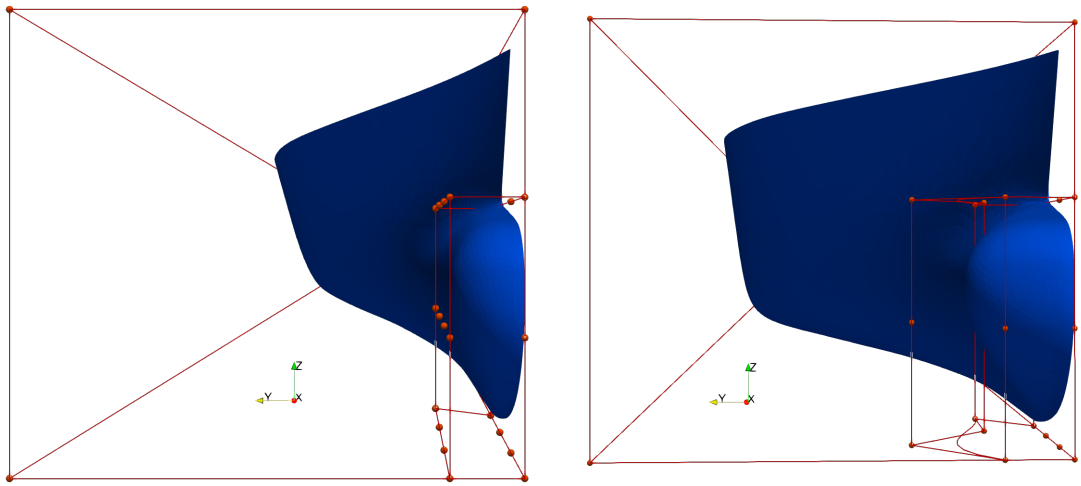
which was translated into a matrix containing the three-dimensional coordinates, and then the module DVGeometry of pyGeo, parametrized the hull. Afterwards, the generation of the FFD box took place, in which the geometry should be fully embedded, and for that reason, two boxes were designed, as shown in Figure 5.2. As mentioned, the bulbous area was of interest making impossible the creation of a box, that enclosed the geometry while being able to handle small deformations because the box is required to be as close to the object as possible. Thus, the generation of a larger (parent) box to ensure the enclosure of the entirety of the hull and a smaller one (child) responsible for the deformation of the object was necessary. The main focus of this chapter's study was to prove the applicability of the shape optimization concept by coupling the two softwares already used; therefore, the boxes were created with the least points possible, with the exemption of the child FFD box in the x and z direction. As shown in table 5.1, three and four points were selected for the z and x directions, respectively. This approach was selected to achieve a smooth result and avoid peculiar shapes during the optimization phase.

**Table 5.1:** *FFD box dimensions.*

| **FFD box** | **Number of points** | | |
|---|---|---|---|
| | nx | ny | nz |
| Parent box | 2 | 2 | 2 |
| Child box | 4 | 2 | 3 |

The final step of the geometry alternation, for the completion, was the displacement of the control points, and as previously, the module DVGeometry was utilized. It provides the option to move all the FFD points together or independently in each direction just by defining the desired value. In order to maintain the simplicity of the examined problem, only one set of points, in the y-direction, was chosen for displacement; as moving from bow to stern a set is defined as the points that have the same x coordinate value. As shown in Figure 5.2, the second set was selected for the transposition, because of its positioning closer to the fore of the bulbous, the part that is primarily responsible for forming the flow around the hull.

The Python script developed through the process described above was then incorporated into the optimization code of Section 4.3 to retrieve the output of the optimizer as an input for the displacement value of the control points and reach an optimum condition. The difference with the trim optimization procedure of Chapter 4 is the way of handling the geometry; instead of applying a rotation, the pyGeo package is called to deform the initial geometry while the remaining process stays the same.



**(a)** *Original geometry* **(b)** *Deformed geometry*

**Figure 5.3:** *The effect of the FFD control points perturbation on the bulbous region.*

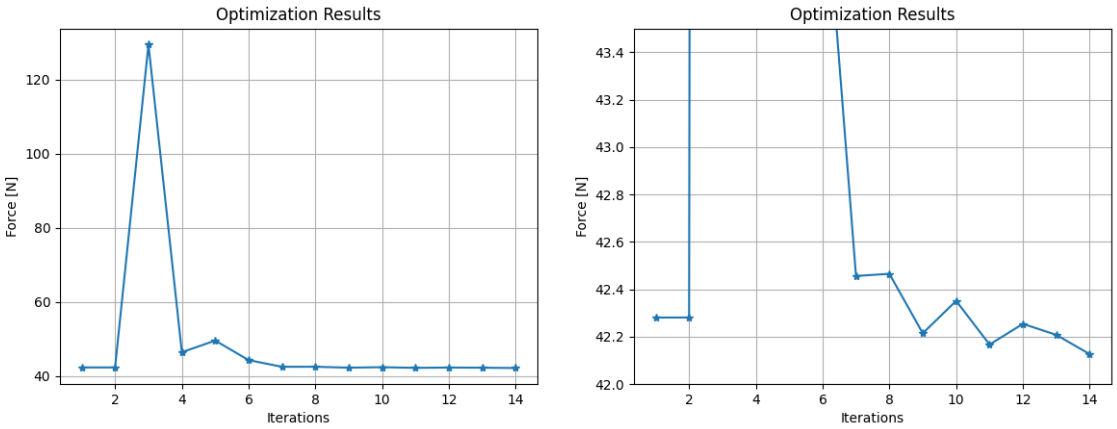
## 5.3 Hull form optimization results

The optimization process concluded in a reduction of the total resistance equal to 0.33%, but the results should be treated with caution. First of all, due to technical difficulties with the electricity in the campus facilities the optimization algorithm did not manage to converge, as shown in Figure 5.4 and in conjunction with the increased computational time needed to acquire the optimum value was decided to present those results instead. The final iterations tend to converge, partially confirming the validity of the resistance improvement. However, the required displacement for the control points is relatively small, suggesting questionable the existence of such a model as it would require advanced manufacturing techniques since the production of such small tolerances is impossible with conventional methods. Thus, the effectiveness of the optimization method is not justified from a financial perspective. Should be noted, however, that the extension of the displacement in more points, only one set of points was displaced, could result in more favourable geometry differences.

Following the layout of Chapter 4, the optimization results are presented, but the differences in the final design are not obvious, so a comparison with the initial geometry was omitted.

**Table 5.2:** *Total resistance for the initial and final condition of the bulbous shape.*

| **Conditon** | **Point Displacement [m]** | **Resistance [N]** | **Difference [%]** |
|---|---|---|---|
| Initial | 0.000 | 42.26 | -0.33 |
| Final | -0.003 | 42.12 | |




**(a)** *The full results history.* **(b)** *Zoomed view of the results history.*

**Figure 5.4:** *The value of total resistance at each shape optimization cycle.*

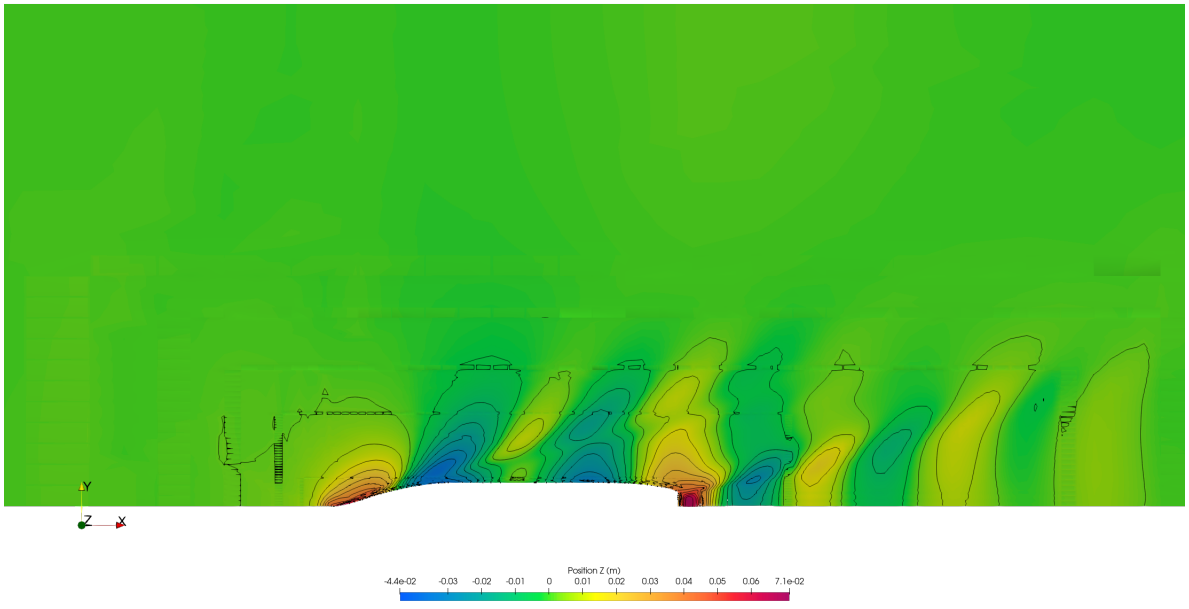
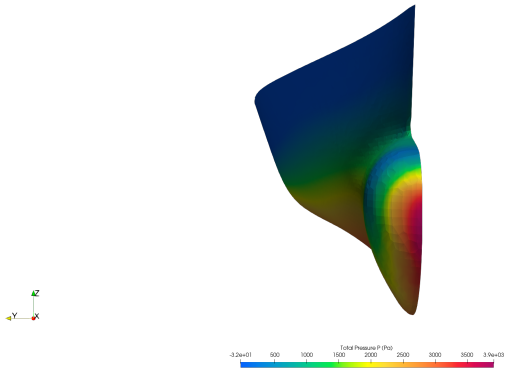
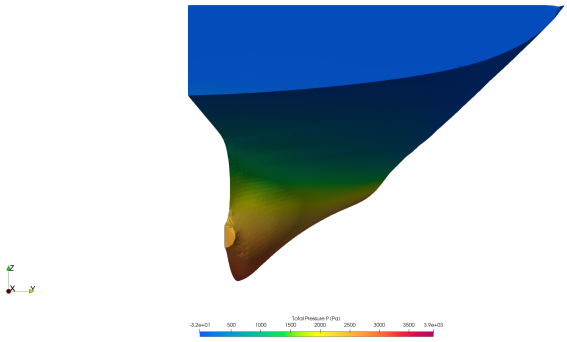


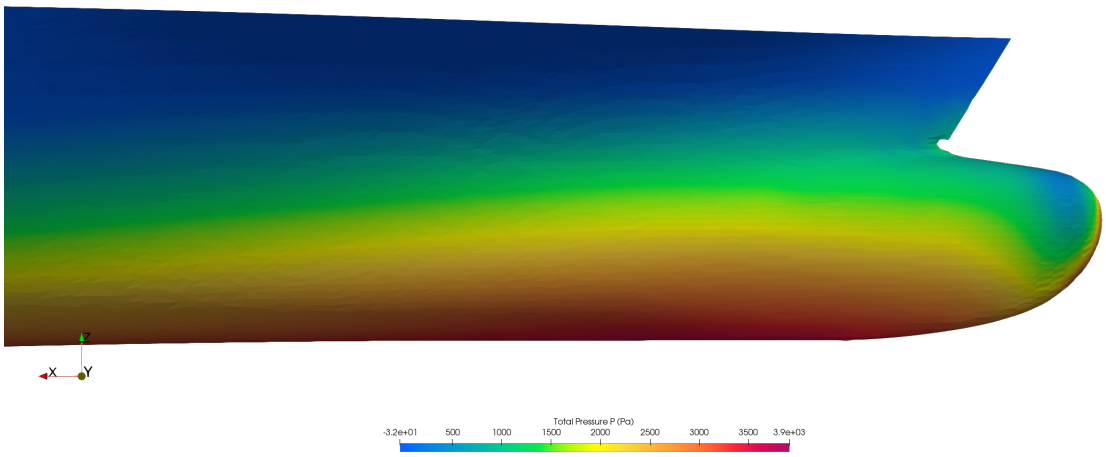

**Figure 5.5:** *Height of the Kelvin waves on the free surface at the final shape optimizaton condition for Froude number 0.26.*



(a) *Stem View*



(b) *Stern View*




(c) *Side view of the bulbous area.*

**Figure 5.6:** *Pressure distribution on the KCS hull at final shape optimization condition for Froude number 0.26, at the bulbous area*

# Chapter 6

# Conclusion

## 6.1 Conclusions

In this thesis, the coupling of the open-source softwares, OpenFOAM and OpenMDAO, was investigated for the problem of trim and hull shape optimization. Generally, the trim optimization concept consists of calculating the ship's resistance for a range of specified trim angles. For tackling the problem, despite its simplicity, an automated process was developed, whose applicability could then be extended for hull form optimization. The findings of the trim optimization were reassuring for the correctness of the developed methodology and managed to decrease the total resistance value by 0.85%. The combination of trim and sinkage was then examined, which showed an even bigger improvement, as expected because the draught reduction has a significant effect on the friction resistance. Finally, the thesis was concluded with the hull form optimization with encouraging results, although the process was not finalized due to technical difficulties. Overall, the initial goal of the thesis was fullfiled while providing opportunities for improvement through future research.

## 6.2 Suggestion for future work

Based on the topic examined in this thesis and the results, several suggestions for future research can be proposed; the summary of which are briefly described below:

- Now that the effectiveness of the concept has been proved, the addition of the vessel's speed in trim optimization can be studied. This would provide a better understanding of the trim effects in different sea states, allowing for more choices during the ship's voyage.

- The simulation process can be enriched by including the appendages of the hull and also by modelling the behaviour of the propeller, to approach as close as possible the real problem. Also, towards this direction, the sea state condition can be modelled and integrated into the simulation, since the bulbous is more effective for reducing the wave-making resistance.
- In engineering the efforts for minimizing the time required for various processes is continuous and very important, as most of the time deadlines must be met. For that reason, a potential research topic would be the investigation of the optimizer's effect on the overall computational time. Another gradient-free optimization algorithm, such as genetic algorithms, can be assessed or even the adjoint method. In addition, the deformation of the mesh grid, based on the updated geometry, can significantly reduce the computational cost because instead of generating the mesh in each iteration, it is created only once at the beginning of the optimization.

Finally, this thesis lays the foundations for an open-source approach of MDO in ship applications, upon which several design variables can be integrated by simulating, for example, the structural integrity and the stability of the ship. In addition, all these variables can simultaneously contribute to the design of the propulsion system to achieve an efficient solution without compromising the effectiveness of the other systems of the ship.

# Bibliography


[1] Anderson, D., Tannehill, J. C., & Pletcher, R. H. (2016). *Computational fluid mechanics and heat transfer*. Taylor & Francis.

[2] Ferziger, J. H., Perić, M., & Street, R. L. (2020). *Computational methods for fluid dynamics*. Springer International Publishing. https://doi.org/10.1007/978-3-319-99693-6

[3] Gao, X., Sun, K., Shi, S., Wu, B., & Zuo, Z. (2019). Research on influence of trim on a container ship's resistance performance. *Journal of Physics: Conference Series*, *1300*(1), 012105. https://doi.org/10.1088/1742-6596/1300/1/012105

[4] Gray, J. S., Hwang, J. T., Martins, J. R. R. A., Moore, K. T., & Naylor, B. A. (2019). OpenMDAO: An open-source framework for multidisciplinary design, analysis, and optimization. *Structural and Multidisciplinary Optimization*, *59*(4), 1075–1104. https://doi.org/10.1007/s00158-019-02211-z

[5] Greenshields, C. J., & Weller, H. G. (2022). *Notes on computational fluid dynamics: General principles*. CFD Direct.

[6] Hansen, H. (2010). Assistance tools for operational fuel efficiency, In *9th conference computer and it applications in the maritime industries (compit), 2010*.

[7] Harvald, S. A. (1992). *Resistance and propulsion of ships*.

[8] Hino, T., Stern, F., Larsson, L., Visonneau, M., Hirata, N., & Kim, J. (Eds.). (2021). *Numerical ship hydrodynamics*. Springer International Publishing. https://doi.org/10.1007/978-3-030-47572-7

[9] Hirt, C., & Nicholls, B. (1981). Volume of fluid (vof) method for dynamics of free boundaries comput. Phy.

[10] Hochkirch, K., & Mallol, B. (2013). On the importance of full-scale cfd simulations for ships, In *11th international conference on computer and it applications in the maritime industries, compit*.

[11] Iakovatos, M., Liarokapis, D., & Tzabiras, G. (2013). Experimental investigation of the trim influence on the resistance characteristics of five ship models. https://doi.org/10.1201/b15813-5

[12] Islam, H., & Soares, C. G. (2019). Effect of trim on container ship resistance at different ship speeds and drafts. *Ocean Engineering*, *183*, 106–115. https://doi.org/10.1016/j.oceaneng.2019.03.058

[13] Issa, R. I. (1986). Solution of the implicitly discretised fluid flow equations by operator-splitting. *Journal of computational physics*, *62*(1), 40–65.

[14] Kenway, G., Kennedy, G., & Martins, J. R. R. A. (2010). A cad-free approach to high-fidelity aerostructural optimization. https://doi.org/10.2514/6.2010-9231

[15] Kim, H., & Yang, C. (2010). A new surface modification approach for CFD-based hull form optimization. *Journal of Hydrodynamics*, *22*(S1), 503–508. https://doi.org/10.1016/s1001-6058(09)60246-8

[16] Larsson, L., Raven, H., & Paulling, J. (2010). *Ship resistance and flow*. Society of Naval Architects; Marine Engineers.

[17] Launder, B. E., & Sharma, B. I. (1974). Application of the energy-dissipation model of turbulence to the calculation of flow near a spinning disc. *Letters in heat and mass transfer*, *1*(2), 131–137.

[18] Lee, J., Yoo, S., Choi, S., Kim, H., Hong, C., & Seo, J. (2014). Development and application of trim optimization and parametric study using an evaluation system (solution) based on the rans for improvement of eeoi. https://doi.org/10.1115/omae2014-24296

[19] Li, J., Duan, W., Chen, J., Ma, S., & Zhang, Y. (2022). A study on dynamic trim optimization of vlcc oil tanker in wind and waves, *253*, 111270. https://doi.org/10.1016/j.oceaneng.2022.111270

[20] Lv, X., Wu, X., Sun, J., & Tu, H. (2013). Trim optimization of ship by a potential-based panel method. *Advances in Mechanical Engineering*, *5*, 378140. https://doi.org/10.1155/2013/378140

[21] Menter, F. R., Kuntz, M., Langtry, R. Et al. (2003). Ten years of industrial experience with the sst turbulence model. *Turbulence, heat and mass transfer*, *4*(1), 625–632.

[22] Moukalled, F., Mangani, L., Darwish, M., Moukalled, F., Mangani, L., & Darwish, M. (2016). *The finite volume method*. Springer.

[23] Patankar, S. V., & Spalding, D. B. (1983). A calculation procedure for heat, mass and momentum transfer in three-dimensional parabolic flows. In *Numerical prediction of flow, heat transfer, turbulence and combustion* (pp. 54–73). Elsevier.

[24] Pope, S. B. (2000). *Turbulent flows*. Cambridge ; New York, Cambridge University Press.

[25] Procedures, I.-R. (2014). Guidelines: Practical guidelines for ship cfd applications. *ITTC Rep*, *7*, 2–18.

[26] Reichel, M., Minchev, A., & Larsen, N. (2014). Trim optimisation - theory and practice. *TransNav, the International Journal on Marine Navigation and Safety of Sea Transportation*, *8*(3), 387–392. https://doi.org/10.12716/1001.08.03.09

[27] Ronzheimer, A. (2004). Shape parameterisation based on freeform deformation in aerodynamic design optimization. *ERCOFTAC Design Optimization: Methods & Applications*, *400*.

[28] Ronzheimer, A. (2006). Prospects of geometry parameterization based on freeform deformation in mdo.

[29] Rusche, H. (2003). *Computational fluid dynamics of dispersed two-phase flows at high phase fractions* (Doctoral dissertation). Imperial College London (University of London).

[30] Schmitt, F. G. (2007). About boussinesq's turbulent viscosity hypothesis: Historical remarks and a direct evaluation of its validity. *Comptes Rendus Mécanique*, *335*(9-10), 617–627.

[31] Sederberg, T. W., & Parry, S. R. (1986). Free-form deformation of solid geometric models. *ACM SIGGRAPH Computer Graphics*, *20*(4), 151–160. https://doi.org/10.1145/15886.15903

[32] Sherbaz, S., & Duan, W. (2014). Ship trim optimization: Assessment of influence of trim on resistance of MOERI container ship. *The Scientific World Journal*, *2014*, 1–6. https://doi.org/10.1155/2014/603695

[33] Shivachev, E., Khorasanchi, M., & Day, A. H. (2017). Trim influence on kriso container ship (kcs): An experimental and numerical study. https://doi.org/10.1115/omae2017-61860

[34] Smith, T. W., Jalkanen, J., Anderson, B., Corbett, J., Faber, J., Hanayama, S., O'keeffe, E., Parker, S., Johansson, L., Aldous, L. Et al. (2015). Third imo greenhouse gas study 2014.

[35] Sun, J., Tu, H., Chen, Y., Xie, D., & Zhou, J. (2016). A study on trim optimization for a container ship based on effects due to resistance. *Journal of Ship Research*, *60*(1), 30–47. https://doi.org/10.5957/josr.60.1.150022

[36] Tahara, Y., Peri, D., Campana, E. F., & Stern, F. (2008). Computational fluid dynamics-based multiobjective optimization of a surface combatant using a global optimization method. *Journal of marine science and technology*, *13*, 95–116.

[37] Tu, H., Xia, K., Zhao, E., Mu, L., & Sun, J. (2023). Optimum trim prediction for container ships based on machine learning. *Ocean Engineering*, *277*, 111322. https://doi.org/10.1016/j.oceaneng.2022.111322

[38] Weller, H., Greenshields, C., & Santos, B. (2018). Openfoam. *OpenCFD Ltd (ESI Group)*.

[39] White, F. M., & Xue, H. (2021). *Fluid mechanics* (Ninth edition). New York, NY, McGraw-Hill Education.

[40] Wilcox, D. C. Et al. (1998). *Turbulence modeling for cfd* (Vol. 2). DCW industries La Canada, CA.

[41] Wu, J., Liu, X., Zhao, M., & Wan, D. (2017). Neumann-michell theory-based multi-objective optimization of hull form for a naval surface combatant. *Applied Ocean Research*, *63*, 129–141. https://doi.org/10.1016/j.apor.2017.01.007