Theodoros Kalimeris

Floating wind turbine optimization via surrogate modelling

Hovedoppgave i European Wind Energy Master (MS Wind)
Veileder: Erin Bachynski-Polić
August 2023

NTNU
Kunnskap for en bedre verden

Theodoros Kalimeris

Floating wind turbine optimization via surrogate modelling

NTNU

Kunnskap for en bedre verden

Floating wind turbine optimization via surrogate modelling

Msc thesis

by

Theodoros Kalimeris

Student number:

5465648	TU Delft
582212	NTNU

Committee and supervisors:

Prof. Dr. Axelle Viré	Chairman & TU Delft supervisor
Prof. Dr. Erin Bachyski-Polić	Committee & NTNU supervisor
As. Prof. Dr. Oriol Colomés Gené	TU Delft committee
Dr. Michiel Zaaijer	TU Delft committee
Dr. Maxime Thys	NTNU committee
Matteo Baudino Bessone	Supervisor
Deepali Singh	Supervisor

Public defence: 31st of August, 2023

EWEM　TUDelft　NTNU

Title picture: the University of MAINE

Abstract

In recent years, the interest in deep-water wind energy projects has drastically increased, driven by the considerable wind resource in deep-water locations. In such locations, floating wind turbines are more economically attractive than bottom-fixed wind turbines. Nevertheless, the floating wind industry faces numerous challenges. One major challenge is the high investment costs involved in the manufacturing of the substructure. Additionally, it is important to maintain the wind turbines' power output within the required standards for grid connection. Therefore, the design and optimization of substructures for floating wind turbines has to account for both capital costs and power quality. To investigate the existing research on the optimization of floating wind turbines, a literature review was conducted. Evidently, there is limited research investigating both cost and power quality optimization. Furthermore, the majority of the optimization techniques presented are accompanied by a computationally expensive substructure analysis. Moreover, in the early stage of the substructure design, a variety of different alternative scenarios are tested, requiring an extensive number of optimizations. Thus, the computational resources required by the optimization become a major concern. One promising approach to address this issue is substituting a computationally expensive analysis with a surrogate model. This solution could significantly increase the computational efficiency of the optimization process.

This thesis investigates the trade-off between substructure capital cost and power quality optimization. To achieve that, a multi-objective optimization workflow is developed for the case of a semi-submersible substructure. In parallel with this task, the optimization's analysis blocks requiring most of the computational resources are identified and substituted with a surrogate model. Two different surrogate model implementation approaches have been examined. The first approach substitutes the optimization's substructure analysis tool (NREL's RAFT) entirely. This approach is called the "End-to-End" approach. The second approach substitutes a specific analysis block of RAFT that computes the floater's hydrodynamic properties. This analysis block requires most of RAFT's computational resources. This approach is defined as the "PyHAMS" approach.

The thesis outcome shows that improving the quality of power output comes with the drawback of increasing the substructure's cost. The Pareto front between these two competing criteria is dominated by designs of long pontoons and reduced chain thickness. This is the most economical approach towards reduced cost and improved power quality. These Pareto designs are mainly driven by the boundaries of the design variables. Additionally, the Pareto designs are affected by constraints on the maximum pitch, maximum surge, and the existence of vertical loads on the mooring line's anchor. Regarding the surrogate model-based optimizations, there is a dramatic reduction in the computational resources required to conduct the optimization. Among the two surrogate model approaches, the "PyHAMS" approach achieves a better accuracy at mimicking the original optimization workflow than the "End-to-End". Instead, the "End-to-End" approach is the most computationally efficient one to conduct the optimization. Nevertheless, when factoring in the computational resources to produce the training dataset, the "PyHAMS" approach is the most computationally efficient one. This is the outcome of utilizing a reduced training dataset compared to one utilized by the "End-to-End" approach.

Preface

This master's thesis is conducted as part of the requirements for graduating from the European Wind Energy Master (EWEM). This master's program is a joint double MSc program between Delft University of Technology (TU Delft) and the Norwegian University of Science and Technology (NTNU). After submitting and defending this thesis, a student obtains the master's degree in Offshore and Dredging Engineering at TU Delft, as well as the master's degree in Technology-Wind Energy at NTNU. The work was conducted in collaboration between the Department of Maritime and Transport Technology at TU Delft, and the Department of Marine Technology at the NTNU. The supervision from TU Delft has been done by Prof. Dr. Axelle Viré, PhD candidate Matteo Baudino Bessone, PhD candidate Deepali Singh, and from NTNU by Prof. Dr. Erin Bachyski-Polić.

I want to express my heartfelt thanks to my supervisors for their invaluable guidance during this challenging period. My supervisors, all of whom were on the other side of Europe, were always available and willing to assist me, regardless of the complexity and the effort required. Their critical thinking, combined with their long university experience, inspired me to develop my own thought process and pushed me to become a better version of myself. Apart from obtaining valuable knowledge, they helped me grow as a person.

I am grateful to my fellow EWEM students. It was an honor to spend two years of my master's degree with such unique and exceptional individuals. In a program full of travelling and rapidly-changing environments, they were the only constant. I will always cherish the moments we shared. The knowledge and skills we acquired during our time at the universities, together with the determination and passion we share for creating a sustainable future, will undoubtedly lead us to achieve great things. I wish all of them the best, and I can't wait to see their impact on the green transition. From 3000 kilometres away, I would like to thank my Greek friends for always being there and believing in me. Thanks to them, returning to Greece always felt like a celebration.

Finally, I would like to take this opportunity to express my sincere appreciation to my parents, brother, and partner. Your unwavering love and trust have given me the strength and motivation to overcome any obstacle that came my way. Thank you for always believing in me, in every stage of my life. Without you, my achievements would have been impossible. I consider myself incredibly fortunate to have you in my life.

<div style="text-align: right">

Theodoros Kalimeris
Delft, August 2023

</div>

Grading note

This Thesis report is conducted at TU Delft and corresponds to the thesis conducted by every student of the European Wind Energy Master (EWEM). According to the Academic Program of the EWEM Master, for the Offshore Wind Energy Track, this research corresponds to 30 ECTS, instead of 45. The remaining 15 ECTS correspond to a pre-thesis conducted at NTNU.

During the pre-thesis, the literature review chapter was written as part of the report. Even though this thesis contains the Literature Review chapter, for completeness, this chapter is already graded during the pre-thesis.

Table of Contents

List of Figures

List of Tables

1 Introduction

In recent years, Europe has been making significant efforts towards achieving the green energy transition. Energy transition promises a fossil fuels-free future and ultimately prevents climate change while ensuring energy security throughout the continent. The most notable way to achieve that is by harnessing renewable power sources, such as solar, hydro and wind power. To this end, the European energy framework target sets the goal of covering 42.5% of the market energy needs from renewable sources by 2030 (European Commission, 2023 [1]).

One of the key players in renewable energy is wind energy, which has evolved into a reliable energy source. Harnessing this source is almost entirely represented by the horizontal axis wind turbine, a structure that utilizes the wind's kinetic energy to generate electricity. A typical modern offshore wind turbine is shown in Figure 1. The main three components that can be distinguished are the rotor, the nacelle and the tower. The rotor consists of the blades, all connected to the hub, and exploits the wind kinetic energy by rotating, based on the principles of aerodynamics. This rotating rotor is connected to a generator by an assembly called the drivetrain. The drivetrain and the generator are housed in the nacelle. The combination of the rotor and the nacelle is also called the Rotor-Nacelle Assembly (RNA). Finally, the tower is the support structure of the RNA, transferring the loads applied on it to the ground (for conventional wind turbines). In the case of offshore wind turbines, like the ones shown in Figure 1, the tower is not fixed on the ground. Instead, it is connected to the substructure, which has the same purpose as the tower but is designed for the harsh offshore environment and the additional wave loads. The substructure can be fixed to the seabed or floating. The connecting piece between the tower and the substructure is called the transition piece.

Figure 1: Offshore horizontal axis wind turbine (left). Offshore horizontal wind turbine sketch (right)

Wind energy has always been a popular renewable source, but especially in recent years, the wind industry is booming. Today's unprecedented Levelized Cost of Energy (LCoE, Figure 2), mainly driven by increasing the wind turbines size and reducing installation cost, makes investing in wind energy more attractive than ever before. Furthermore, recent wind turbines have emerged as the most eco-friendly renewable energy source, with the shortest energy payback time compared to other sources (Lazard report, 2021 [2]). What is more, modern wind turbines are approximately 90% recyclable (according to Ørsted [3]). In the following years, this rate is expected to increase further. There are numerous undergoing projects investigating the existing non-recyclable components of a wind turbine, with an emphasis on the blades. For instance, in March 2022, the first 62 m. fully-recyclable wind turbine blade was produced as part of the Zero wastE Blade ReseArch (ZEBRA) project [4].

Wind farms can be developed both on land and offshore. However, as wind turbines are gradually increasing in size, offshore projects are getting more attention. This is a natural choice, as the offshore wind environment offers a higher wind potential. Constructing in the offshore environment also decreases the environmental impact caused by the project. Specifically, an offshore wind farm results in reduced visual disruption and impact on the local marine ecosystem, compared to an onshore wind farm. Moreover, the offshore environment involves an abundance of space. Despite these benefits, offshore wind energy comes along with many challenges related to deep water installations, Operation and maintenance (O&M), harsh environmental conditions etc. To date, almost every offshore wind farm is built in shallow water depth, with the dominant support structure being the monopile, accounting for 81% of the total number of substructures (Wind Europe [5]). More generally, fixed-bottom foundations cover 99.9% of offshore wind substructures. Nevertheless, the undergoing research in floating offshore wind is booming, as there are areas of very high wind potential but depths greater than 100 m.
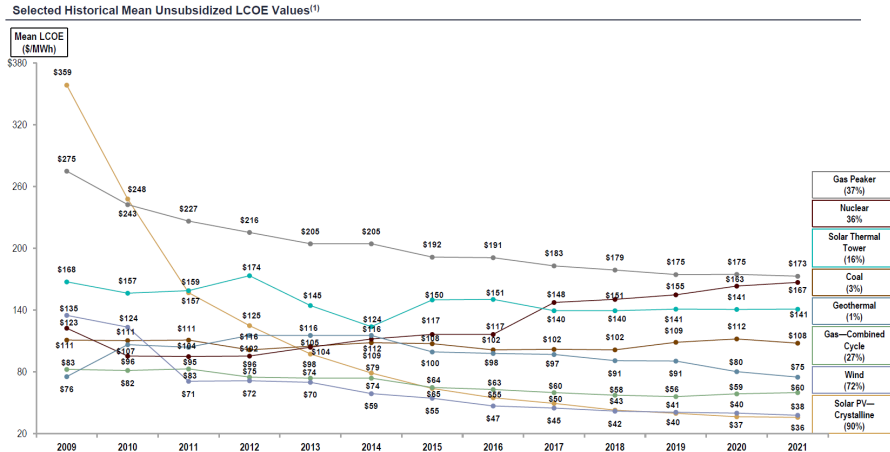
Figure 2: LCOE evolution of different energy sources (Hamilton, 2021 [2])

The current economic and industrial limitations of fixed bottom structures for deep water make floating platforms the only feasible solution for deep water wind farms. However, there are significant challenges to overcome before the floating wind industry can be considered mature. Currently, there is a limited number of floating wind farms developed. In Europe, the are four fully commissioned floating wind farms. These are the 30 MW Hywind Scotland, 88 MW Hywind Tampen, 25 MW WindFloat Atlantic and the 50 MW Kincardine wind farm. Hywind Scotland was the first floating wind farm ever developed, commissioned in October 2017.

There are three main platform types, being the spar-buoy, semisubmersible (SS) and tension leg platform (TLP), the overview of which is shown in Figure 3. The main difference between them is the principles followed to stabilize the floater. Starting from the spar, this hollow structure is filled with ballast up to a certain level, keeping the center of mass lower than the buoyancy center, ensuring stability. Additionally, catenary mooring lines are installed, so that the floater stays in place. Although ballast is also used in SS floaters, these types of structures also achieve stability based on the waterplane area's moment of inertia of the structure. For that reason, a SS consists of columns placed further away from the center of the system. Finally, TLP stability is achieved by utilizing the excessive buoyancy of the floater, combined with a tension leg mooring system. Each one of these structures has its own strong and weak points, most of them related to potential depth ranges, manufacturing cost, installation methods etc. There are also new upcoming projects under development, each one of them holding its own advantages. As a reference, such concepts are the Stiedstal Tetraspar (Borg, 2020 [6]) or the Hexafloat concept (Ghigo, 2020 [7]).

Overcoming the existing obstacles is vital for floating wind energy to grow. Most notably, manufacturing and installing floating wind platforms explodes the investment's cost. One of the primary reasons for this is the size of the floater. Even by only considering the raw material needed, floating platforms typically require larger amounts, compared to fixed bottom substructures. Another reason is that all of the current floating wind projects are pilot ones. As a result, floating wind

lacks large-scale component manufacturing, which reduces cost. Moreover, defining the optimal platform-mooring system design, leading to a reduced cost, is a challenging and time-consuming process.

In order to find the ideal balance between all the desired criteria, the substructure needs to be optimized. At first, an optimization was based on the experience of the engineer to understand the way design changes affect a structure, and the changes made were not methodical. Today, however, optimization has evolved to be a structured process. That doesn't mean that the user's experience is not important, as decision-making is still part of the process. The fundamentals of optimizations are further discussed in Section 2.

Optimizing the design of a floating wind turbine platform is crucial for improving the properties of the floating wind turbine based on the unique characteristics of its location. Such properties can be the cost of the substructure, the fatigue accumulated in the wind turbine or the quality of the power produced. Nevertheless, substructure optimization is a process that requires extensive computational resources. These resources are needed in order to simulate the dynamic behaviour of different design configurations.
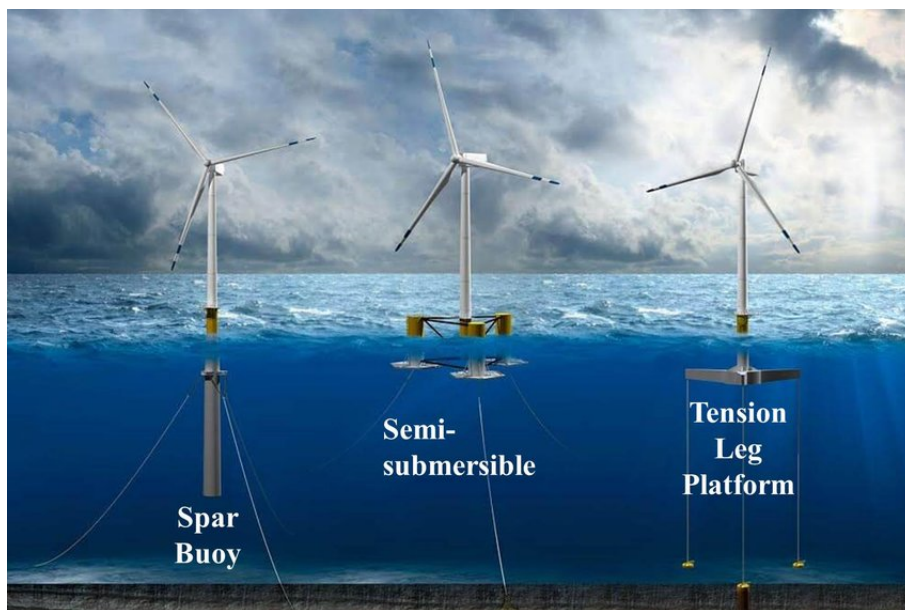
Figure 3: Main floating offshore wind platform types. Illustrated by the NREL [8].

2 Theory

2.1 Hydrodynamics

In the following sections, the theoretical aspects behind the modelling approaches adopted for floating wind turbines are summarized. Those approaches involve the hydrodynamics, aerodynamics, moorings and control of floating wind turbines.

In this section, the basic hydrodynamic-related concepts involved in this thesis are demonstrated, being linear potential flow theory and rigid floating structures motions. The following theory is based on Faltinsen's book on sea loads (Faltinsen, 1993 [9]).

2.1.1 Potential flow theory

Potential flow theory is used extensively in this thesis research and is a simplified and accurate method to calculate the forces and the potential flow field around a structure interacting with a fluid. This problem is shown in Figure 4. This method is applied under the assumptions that:

- The fluid is inviscid, meaning that it has zero viscosity. Thus, the viscous stresses are neglected.

- The fluid is incompressible, which means in practice that the defined fluid volume does not change. Mathematically:

$$\nabla \cdot \boldsymbol{V} = 0 \tag{1}$$

  with

$$\nabla \cdot \boldsymbol{V} = \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} \tag{2}$$

  Additionally, the sum of total flux entering and leaving the control volume is zero.

- The fluid flow is irrotational, which means that the fluid does not rotate and as a consequence, the vorticity (ω) is zero.
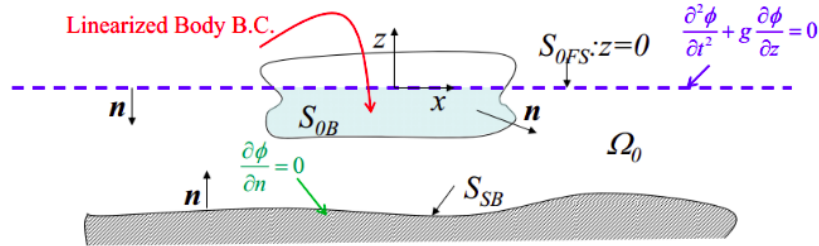
Figure 4: Floating body-water interaction problem (Faltinsen, 1993 [9])

As a consequence to the above mentioned assumptions, the velocity vector can be represented by a scalar function, ϕ. The governing equation of the problem becomes:

$$\nabla^2 \phi = 0 \quad \text{in } \Omega_0 \tag{3}$$

The problem is also defined by the following conditions:

- Sea bottom impermeability boundary condition:

$$\boldsymbol{V} \cdot \boldsymbol{n} = 0 \Rightarrow \nabla \phi \cdot \boldsymbol{n} = 0 \Rightarrow \frac{\partial \phi}{\partial n} = 0 \quad \text{on } S_{SB} \tag{4}$$

- Solid body impermeability boundary condition:

$$\frac{\partial \phi}{\partial n} = \boldsymbol{V_B} \cdot \boldsymbol{n} \text{ on } S_B \tag{5}$$

- Fluid particles on the free-surface boundary condition:

$$\frac{D(z-\zeta)}{Dt} = 0 \rightarrow \frac{\partial \phi}{\partial z} = \frac{\partial \zeta}{\partial t} + \frac{\partial \phi}{\partial x}\frac{\partial \zeta}{\partial x} + \frac{\partial \phi}{\partial y}\frac{\partial \zeta}{\partial y} \quad \text{on } z = \zeta(x,y,t) \tag{6}$$

- Pressure on the free surface is equal to ambient pressure:

$$g\zeta + \frac{\partial \phi}{\partial t} + \frac{1}{2}\left[\left(\frac{\partial \phi}{\partial x}\right)^2 + \left(\frac{\partial \phi}{\partial y}\right)^2 + \left(\frac{\partial \phi}{\partial z}\right)^2\right] = 0 \text{ on } z = \zeta(x,y,t) \tag{7}$$

The equations of the boundary conditions presented are non-linear. In order to linearize the problem, it is assumed that the measured non-linearities are identified by a parameter ϵ. It is additionally assumed that the body motions and ϵ are small around the mean free surface ($z = 0$). Considering those assumptions, ϕ and ζ can be analyzed based on a perturbation method as:

$$\phi = \tilde{\phi}_1 \varepsilon + \tilde{\phi}_2 \varepsilon^2 + \tilde{\phi}_3 \varepsilon^3 + \dots \tag{8}$$

and

$$\zeta = \tilde{\zeta}_1 \varepsilon + \tilde{\zeta}_2 \varepsilon^2 + \tilde{\zeta}_3 \varepsilon^3 + \dots \tag{9}$$

By substituting Equation 8 and Equation 9 into Equation 3 to Equation 7, all non-linear terms become proportional to ϵ, ϕ and ζ. Thus, the potential flow theory problem becomes linear. Particularly for Equation 6 and Equation 7, since ϵ is assumed small, ϵ^n (with $n > 1$) is considered negligible. That leads to:

$$\frac{\partial \phi}{\partial z} = \frac{\partial \zeta}{\partial t} \text{ for } z = 0 \tag{10}$$

and

$$g\zeta + \frac{\partial \phi}{\partial t} = 0 \text{ for } z = 0 \tag{11}$$

2.1.2 Equations of motion

To model the response of a floating structure, it is assumed that the body is rigid with 6 degrees of freedom (DOF), being surge, sway, heave, pitch, roll, yaw (in the order from 1 to 6), as illustrated in Figure 5, for the case of a floating wind turbine. The equation describing these motions is Newton's second law:

$$\sum_{k=1}^{6} M_{jk}\ddot{\eta}_k(t) = F_j(t) \quad j = 1,\dots,6 \tag{12}$$

where j indicates the excitation force direction, k the DOF motion and η is the DOF motion amplitude as a function of time. M represents the mass matrix of the floater and F_j is the external force or moment. In linear potential flow theory, F_j can be calculated as the sum of hydrostatic pressure integrated over the instantaneous position (S_S) of the body (considering still water), and dynamic pressure integrated over the mean position (S_{0z}). Hence:

$$F_j(t) = \int_{S_{0z}} -\rho\frac{\partial \phi}{\partial t}\boldsymbol{n}dS + \int_{S_s} -\rho gz\boldsymbol{n}dS \quad j = 1,\dots,6 \tag{13}$$
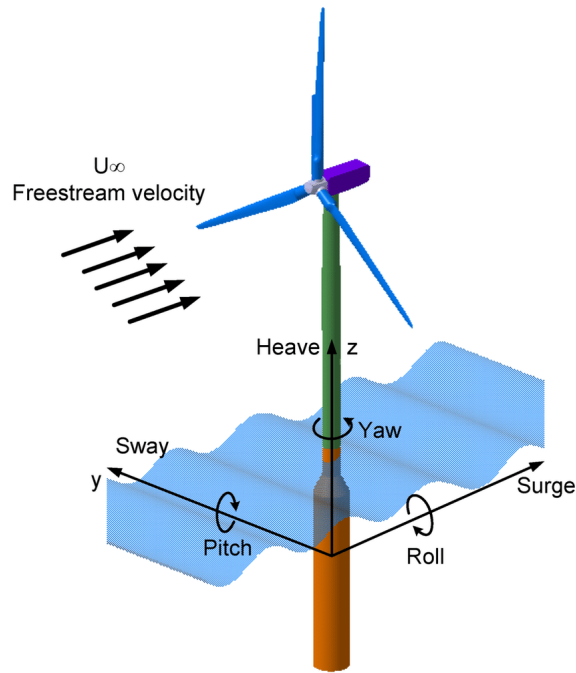
Figure 5: Degrees of freedom for a floating wind turbine platform (Leimeister, 2017 [10])

2.1.3 Radiation and diffraction problem

In linear potential flow theory, the wave-structure interaction problem can be divided into two sub-problems, *diffraction* and *radiation*. Diffraction refers to the case where the body is stationary inside the wave field, based on which the wave excitation loads are calculated. Radiation refers to the situation where the body oscillates in a field without incoming waves, based on which the added mass, damping, and stiffness of the system are calculated. This principle is depicted in Figure 6.

Based on this approach, the velocity potential ϕ can be expressed as:

$$\phi(x, y, z, t) = \phi_0(x, y, z, t) + \phi_D(x, y, z, t) + \phi_R(x, y, z, t) \tag{14}$$

With ϕ_0, ϕ_D and ϕ_R referring to the potential flow resulting from the incident, diffracted and reflected waves, respectively.
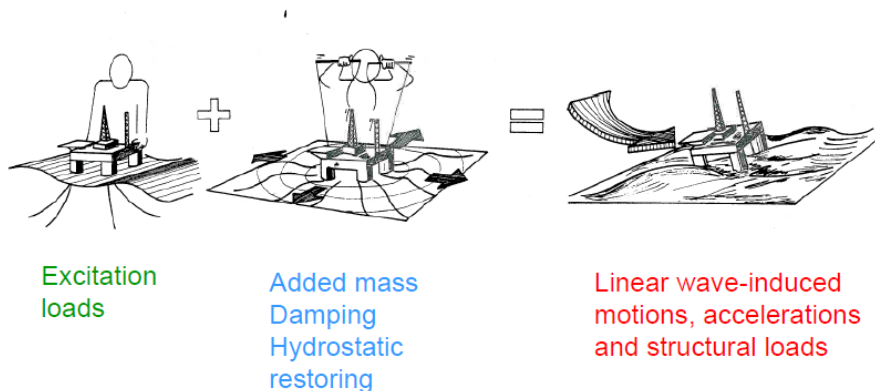
Figure 6: Diffraction and radiation problems demonstration in linear potential flow theory (Faltinsen,1993 [9])

In Equation 14, ϕ_0 and ϕ_D represent the diffraction problem's potential flow, while ϕ_R represents the radiation problem. For the diffraction case, integration of ϕ_0 and ϕ_D along the mean submerged hull surface gives the excitation forces of the problem. Mathematically this is given by:

$$F_{exc,k}(t) = -\int_{S_{0B}} \rho \frac{\partial \phi_0}{\partial t} n_k dS - \int_{S_{0B}} \rho \frac{\partial \phi_D}{\partial t} n_k dS, \text{ for } k = 1, ..., 6 \tag{15}$$

Equation 15 splits the excitation force into two terms. The first term (integration of ϕ_0) corresponds to the hydrodynamic loads applied from the flow to the body as if the body was not present. These loads are called the *Froude-Kriloff loads*. The second term (integration of ϕ_D) corresponds to additional loads applied from the flow to the body due to the presence of it. The body's presence diffracts the flow around it, causing the *diffraction loads*.

For the radiation problem, the freely moving body generates (radiation) waves. These waves and the body's response result in hydrodynamic loads to the body, which are identified as *added mass (A), waves' radiation damping (B)* and *restoration (C)* terms. The restoration term is related to the hydrostatic forces applied on the instantaneous position of the body. This is described by the second term of Equation 13. The added mass and waves' radiation damping account for the additional volume of water reacting to the acceleration/deceleration of the submerged part of the structure. They are calculated by integrating ϕ_R along the mean submerged hull surface, given by:

$$F_{rad,k}(t) = -\int_{S_{0B}} \rho \frac{\partial \phi_R}{\partial t} n_k dS, \text{ for } k = 1 \dots 6 \tag{16}$$

By analyzing Equation 16 terms into loads proportional to the body velocity and acceleration, F_{rad} is given by:

$$F_{rad,k}(t) = \sum_{j=1}^{6} \left\{ -A_{kj} \ddot{\eta}_j - B_{kj} \dot{\eta}_j \right\}, \text{ for } k = 1 \dots 6 \tag{17}$$

where A_{kj} and B_{kj} are the added mass and damping coefficients, respectively. For a given response frequency, these coefficients are calculated as:

$$A_{kj}(\omega) = \Re \left[\rho \int_{S_{0B}} \varphi_j n_k dS \right] \tag{18}$$

and

$$B_{kj}(\omega) = -\omega \widetilde{\mathfrak{S}} \left[\rho \int_{S_{0B}} \varphi_j n_k dS \right] \tag{19}$$

2.1.4 Strip theory

An alternative approach to compute the hydrodynamic properties A, B and F is via *strip theory*. The basis of this approach is that, for slender (elongated) bodies, it can be assumed that the fluid motion problem can be approximated as a 2D problem.

In strip theory, the submerged body is divided into strips, as shown in Figure 7. Following the division, the two-dimensional added mass and waves' radiation damping coefficient are calculated for each strip. The total added mass and hydrodynamic damping of the body is approximated as the summation of the added mass and hydrodynamic damping of each strip. The same philosophy is followed to calculate the total excitation force, which is the summation of the excitation forces of each strip. Moreover, the application of strip theory implies that the flow variations in the longitudinal direction of the body are insignificant compared to the cross-directional plane. This assumption results in significant errors at the edges of the body.

There are limited applications of strip theory in this thesis, as the error introduced by the method makes its estimations unreliable. Nevertheless, the formulas utilized to calculate added mass, hydrodynamic damping and hydrodynamic forces based on strip theory can be found in Section 4.3.
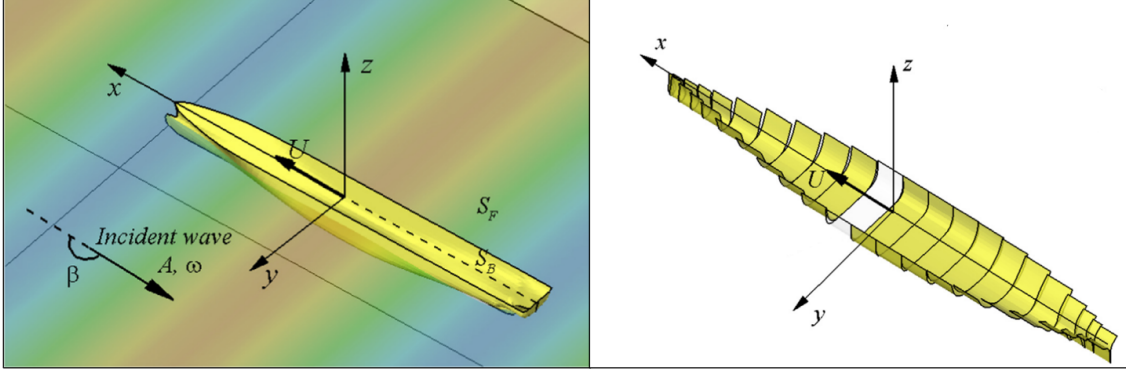
Figure 7: Application of Strip theory (Park, 2016 [11])

2.1.5 Frequency domain analysis

Based on Section 2.1.3, F_j can be written as:

$$F_j(t) = \sum_{k=1}^{6} F_j^{exc}(t) - A_{jk}(\omega)\ddot{\eta}_k(t) - B_{jk}(\omega)\dot{\eta}_k(t) - C_{jk}\eta_k(t) \quad j = 1, \ldots, 6 \tag{20}$$

where A is the hydrodynamic added mass matrix, B is the damping matrix, and C is the stiffness matrix. Dots denote derivation over time ($\frac{d}{dt}$). Among them, A and B matrices depend on the oscillation frequency of the degree of freedom they refer to. Based on that, Equation 12 is written as:

$$\sum_{k=1}^{6} \left(M_{jk} + A_{jk}(\omega) \right) \ddot{\eta}_k(t) + B_{jk}(\omega)\dot{\eta}_k(t) + C_{jk}\eta_k(t) = F_j^{exc}(t) \quad j = 1, \ldots, 6 \tag{21}$$

or, in matrix form:

$$(\mathbf{M} + \mathbf{A})\ddot{\eta} + \mathbf{B}\dot{\eta} + \mathbf{C}\eta = \mathbf{F} \tag{22}$$

Another important consequence of linearity is the validity of the superposition principle. Due to that, for a linear irregular wave field, the irregular wave observed can be divided into regular linear harmonic waves with different wave amplitudes and frequencies. Thus, the irregular wave excitation forces are equal to the summation of harmonic excitation forces. Assuming that such an excitation force has an amplitude $|\xi_a X|$ and frequency ω, it can be written as:

$$\mathbf{F}(t) = \mathcal{R}\left\{ \xi_a \mathbf{X}(\omega, \beta) e^{i\omega t} \right\} \tag{23}$$

With β being the angle between the wave (which is responsible for the loads) and the x-axis, according to Figure 5. Since the response is proportional to the excitation amplitude (for a certain ω), it can be expressed as:

$$\boldsymbol{\eta}(t) = \mathcal{R}\left\{ \boldsymbol{\eta}_a e^{i\omega t} \right\} \tag{24}$$

By substituting Equation 23 and Equation 24 in Equation 22, the result obtained is:

$$\left(-\omega^2(\mathbf{M} + \mathbf{A}(\omega)) + i\omega\mathbf{B}(\omega) + \mathbf{C}\right)\boldsymbol{\eta}_\alpha = \xi_a\mathbf{X}(\omega, \beta) \tag{25}$$

Based on the linear relation between ξ_a and η_a, the *Response Amplitude of Operation* (RAO) can be defined:

$$RAO = |\eta_\alpha/\xi_a| \tag{26}$$

RAO is a convenient expression to describe the effect that a unit force amplitude has on the degrees of freedom amplitude. Based on it, it is easy to understand the resulting amplitude for a specific force and also to compare the effect of different excitation forces.

2.2   Aerodynamics

In this section, the fundamentals of aerodynamics are analyzed, in order to present frequently used terms throughout the thesis. These concepts can all be explained by demonstrating the 1D momentum theory, which is the simplest approach to calculating the rotor thrust force and power output. The following paragraphs are in line with Hansen's book on wind turbines' aerodynamics (Hansen, 2015 [12]).

To derive the thrust force and power output from the 1D momentum theory, it is assumed that:

- The flow is homogeneous, incompressible and steady-state.

- The air is inviscid (no frictional forces, no energy dissipation).

- The wind turbine rotor is modelled as a permeable disk.

- Right upstream of the rotor, a pressure rise takes place, and right downstream of the rotor, a pressure drop takes place. Thus, there is a pressure drop at the rotor plane.

- There is no flow velocity discontinuity.

- Far from the rotor plane, pressure is ambient (atmospheric).

According to these assumptions, the flow streamlines form the control volume shown in Figure 8.
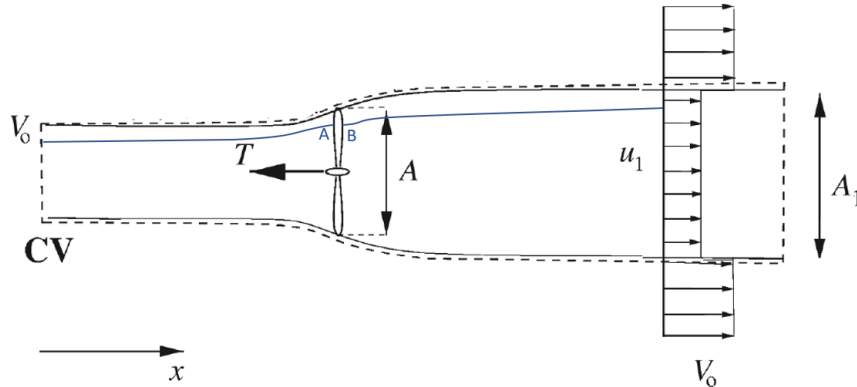
Figure 8: 1D momentum theory control volume. (Hansen, 2015 [12])

Since there is no air input or output from the sides of the control volume, the conservation of mass can be written as:

$$\rho A_0 v_0 = \rho A_1 v_1 = \dot{m} \tag{27}$$

where v_0 and v_1 are the inlet and outlet velocities, respectively, ρ is the air density, and A corresponds to the plane area. Also, the change in the fluid momentum occurs from the rotor-flow interaction caused by the thrust force. By applying the equilibrium of momentum, the thrust force, T, is equal to:

$$T = v_0 \left(\rho A_0 v_0 \right) - v_1 \left(\rho A_1 v_1 \right) \tag{28}$$

By substituting Equation 27 into Equation 28, T is given by:

$$T = \dot{m} \left(v_0 - v_1 \right) \tag{29}$$

According to the assumptions, there is a pressure discontinuity at the rotor plane. By defining two points A and B right before and after of the rotor plane, respectively, the Bernoulli theorem is applied for each streamline, giving:

$$p_0 + \frac{1}{2} \rho v_0^2 = p_A + \frac{1}{2} \rho v_A^2 \tag{30}$$

from the inlet until point A and:

$$p_B + \frac{1}{2} \rho v_B^2 = p_1 + \frac{1}{2} \rho v_1^2 \tag{31}$$

from point B until the outlet. With the pressures p_A and p_B defined, T can also be calculated as the reaction to the pressure difference caused by the rotor times the area of the rotor. According to the flow velocity continuity assumption, $v_A = v_B$. Additionally, due to the ambient pressure assumption $p_0 = p_1$. Based on those remarks:

$$T = A(p_A - p_B) = A(\frac{1}{2}\rho v_0^2 - \frac{1}{2}\rho v_1^2) \tag{32}$$

or

$$T = \frac{1}{2}\rho A \left(v_0^2 - v_1^2 \right) \tag{33}$$

Equation 29 and Equation 33 are equal, which means that:

$$\dot{m} \left(v_0 - v_1 \right) = \frac{1}{2}\rho A \left(v_0^2 - v_1^2 \right) \tag{34}$$

Since $\dot{m} = \rho A v_A$, utilizing Equation 34 leads to:

$$\frac{1}{2}\rho A \left(v_0^2 - v_1^2 \right) = \rho A v_A \left(v_0 - v_1 \right) => v_A = \frac{1}{2} \left(v_0 + v_1 \right) \tag{35}$$

Equation 35 simply demonstrates that, in the 1D momentum theory, the wind velocity before the rotor is equal to the average of the inlet and outlet velocity.

By introducing an induction factor, α, for the velocity reduction from the inlet until point A:

$$\alpha = \frac{v_0 - v_A}{v_0} \tag{36}$$

v_A and v_1 are calculated as:

$$\begin{aligned} v_A &= v_0(1 - \alpha) \\ v_1 &= v_0(1 - 2\alpha) \end{aligned} \tag{37}$$

By observing Equation 37, it becomes evident that the induction factor makes physical sense for values within the range of [0, 0.5]. If $\alpha < 0$, then $V_1 > V_0$, meaning that the occurrence of an obstacle in the flow results in a wind speed increase (without any pressure drop). If $\alpha > 0.5$, then the outlet wind speed becomes negative. In this case, the stream tube assumption breaks down. Regarding power production, with the assumption that the flow is frictionless, only the rotor is able to extract energy from the system. This energy is equal to the work done by the rotor:

$$P = Tv_A = \frac{1}{2}\rho A \left(v_0^2 - v_1^2\right) v_A \tag{38}$$

By utilizing Equation 37:

$$P = \frac{1}{2}\rho A v_0^3 4\alpha(1 - \alpha)^2 \tag{39}$$

Converting Equation 38 to Equation 39 is very handy for understanding the efficiency of a wind turbine. The total energy available for a specific wind speed is equal to the wind kinetic energy, being:

$$P_w = \frac{1}{2}\rho A v_0^3 \tag{40}$$

By dividing P with P_w, the efficiency of a wind turbine can be expressed by the power coefficient C_p:

$$C_P = P/P_w = 4\alpha(1 - \alpha)^2 \tag{41}$$

Equation 41 is a function of α. Apart from C_p, there is an additional coefficient that is widely used in wind energy engineering. The first one is the thrust coefficient, C_T, which corresponds to the thrust force applied to the rotor, divided by the maximum force that the moving wind can apply to the rotor's swept area. Mathematically, C_T is given by:

$$C_T = \frac{T}{\frac{1}{2}\rho_{air}AV^2} \tag{42}$$

where A is the rotor swept area and V is the average wind speed at hub height.

2.3   Catenary mooring system

Several models are available for catenary mooring modelling, each with unique advantages and varying levels of fidelity. In this thesis, the mooring lines are analyzed based on a quasi-static analysis. This approach calculates the mooring loads by conducting a static analysis for a given dynamic response (DNV-GL-OS-E301, 2015 [13]). In other words, for a given response of a floating structure, the loads applied by the mooring line are calculated as if the structure was in static equilibrium. Even though this approach offers reduced accuracy compared to a dynamic analysis

model, the computational time reduction achieved makes quasi-static models very popular in optimization processes.

According to the quasi-static approach, the horizontal and vertical components of the mooring line tension, F_x and F_z respectively, on a given point on the mooring line, are calculated by solving the static system shown in Figure 9 of a suspended line attached to a floating structure. Hence, F_x is given by solving Equation 43:

$$z = \frac{F_x}{w_{water}} \left(\cosh \left(\frac{w_{water}}{F_x} x \right) - 1 \right) \tag{43}$$

where z is the point's vertical distance from the seabed, w_{water} is the submerged weight of the mooring line per meter, and x is the horizontal projection of the distance between the point and the mooring line's touchdown point. For the case of the fairlead point, which is the main point of interest, z corresponds to the fairlead height, and x to the horizontal distance between the touchdown and the fairlead point.

The vertical component F_y is given as:

$$F_y = w_{water} S \tag{44}$$

where S is the length of the suspended mooring line. With both F_x and F_y, the total mooring line tension force, F_T, is calculated by applying the Pythagorean theorem, resulting in:

$$F_T = \sqrt{{F_x}^2 + {F_y}^2} \tag{45}$$
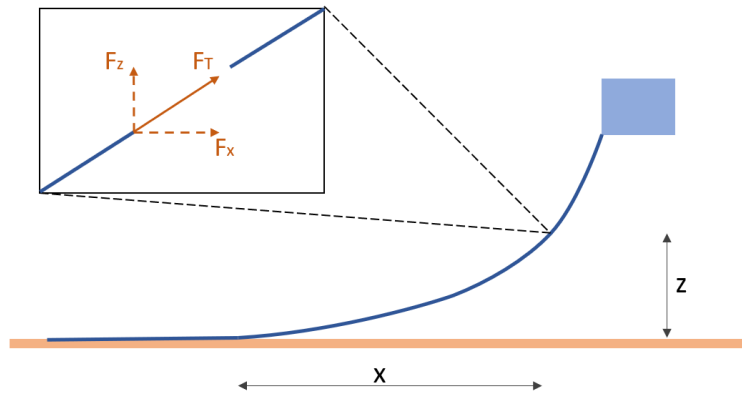
Figure 9: Mooring static analysis problem

2.4 Controlling

In wind energy, controlling consists of the systems responsible for ensuring that the wind farm, wind turbines (individually) and the subsystems operate within a desirable range. Focusing only on an individual wind turbine, the controller:

- Yaws the nacelle.

- Starts and stops the turbine.

- Ensures that the rotor rotation, generator torque and power output stay within the design limits.

In the following paragraphs, the fundamentals of modern controlling techniques are analyzed, with an emphasis on those applied to this thesis. Similarly to the aerodynamics section (Section 2.2), the theory given below is based on Hansen's book on wind turbines aerodynamics (Hansen, 2015 [12]).

It is important to mention, though, that the above-mentioned functions are not only affected by the individual controlling of the turbine but additionally by the wind farm controller. For instance, the yaw control can be used to misalign the rotor from the flow direction, in order to reduce the wake effects felt from the downstream placed turbine. For this thesis, however, farm control is not considered. Specifically, only pitch and torque control are considered, analyzed in Section 2.4.1.

2.4.1 Pitch and torque control

As wind turbines become larger, their controlling systems become more complicated and vital. Modern controllers are highly sophisticated, and apart from power production, they ensure the structural integrity of the turbines. This is succeeded by reducing the loads applied on the wind turbine. Most of today's wind turbine controllers are based on variable speed (VS) and variable pitch (VP). Variable speed control enables the wind turbine to produce power for a variety of different rotor (angular) velocities. This is achieved by the development of power electronics. In order for the turbine to operate at the most efficient rotor speed (for a given wind speed), generator torque control is utilized. Variable pitch allows the turbine to maintain power constant while reducing the loads applied to the wind turbine's blades. This is achieved by acting on the pitch angle of the wind turbine blades (pitch control).

The way that those two systems collaborate depends on the operational region of the wind turbine. Typically, four regions are identified, depicted in Figure 10. The regions' typical characteristics and the controlling strategies used are:

- Region I: Corresponds to wind speeds lower than the cut-in wind speed, which is the minimum wind speed needed to extract power. The cut-in wind speed is defined by the wind turbine manufacturer. For wind velocities in Region I, the wind turbine is typically idling or parked.

- Region II: Defined between the cut-in and rated wind speed. In this region, the objective of the controller is to maximise power production by tracking the optimum C_p, in order to extract as much energy as possible from the wind. For that reason, the tip speed ratio (TSR), λ, and the pitch angle of the blades, θ, stay constant (at the optimum angle). On the contrary, the rotor angular velocity and the generator torque increase with the wind speed (linearly and quadratically).

- Region III: This region extends from the rated until the cut-off wind speed. When operating inside this region, the wind turbine power is kept constant at its rated value. The thrust force applied on the rotor is reduced with the increasing wind speed. A typical diagram of thrust force versus wind speed can be found in Figure 11. In this region, both variable speed and variable pitch are applied simultaneously. The pitch angle increases with wind speed, as observed in Figure 10.

- Region IV: For wind speeds higher than the cut-out speed, the wind turbine blades feather (pitch at 90 degrees) and the wind turbine is idling or parked, producing no power.

In the following paragraph, the way that both the torque and pitch controllers function is analyzed.
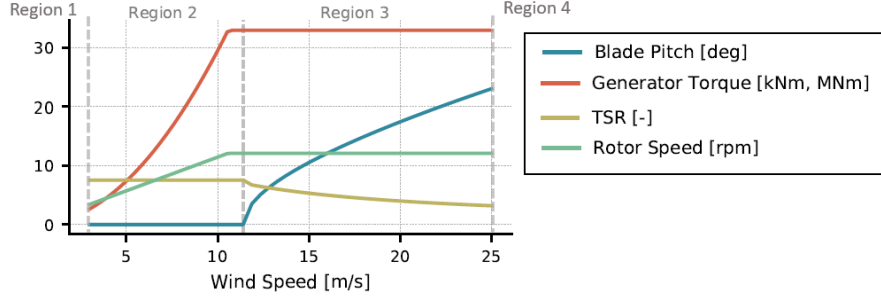
Figure 10: Typical diagram of operational regions of a wind turbine (Abbas, 2022 [14])

2.4.2 Torque control operation

As already stated, torque control is always active during power production (Regions II and III). Starting from below-rated wind speeds, the power coefficient C_p and tip-speed ratio λ are constant and optimum. By utilizing this information, the torque control is able to adapt torque, Q, based on the rotor angular velocity, ω. Analytically, generator torque is calculated as:

$$Q = \frac{P}{\omega} = C_p \frac{1}{2} \rho \pi R^2 V_0^3 \tag{46}$$

where R is the rotor radius and V_o is the wind speed. In practice, the only term of Equation 46 that is not easily and accurately estimated is the wind speed on the rotor, while the rest of Equation 46 terms are constants for below-rated conditions. What is easier to monitor, though, is the rotor rotational velocity ω, which is given by:

$$\omega = \frac{V_o \lambda}{R} \tag{47}$$

Substituting Equation 47 into Equation 46 results in:

$$Q = \frac{C_p \frac{1}{2} \rho \pi R^5 \omega^2}{\lambda^3} \tag{48}$$

which is only ω dependent.

For above-rated wind conditions, the torque control collaborates with the pitch control to ensure that power and rotor rotation stay constant. In this region, the torque control functions by preserving either rated power or rated generator torque. Among those two options, the former is the most common one used, and the latter can be found in floating wind turbines. In this thesis, the controller used preserves generator power.

2.4.3 Pitch control operation

Unlike torque control, pitch control is activated mostly in Region III. One obstacle raised by only operating in Region III is that, due to the wind speed fluctuations, the controller should be able to react to the instant rotor rotational speed, even when it is lower than the rated one, for a Region III average wind speed. This is solved by applying Proportional-Integral (PI) pitch control. PI control calculates the pitch angle change needed, Δ_θ by utilizing Equation 49:

$$\Delta_\theta = K_P \Delta_\omega + K_I \int_0^t \Delta\omega(\tau)d\tau \tag{49}$$

In Equation 49, Δ_θ is calculated as the summation of two terms. The first term is equal to the error between the current shaft speed and the rated Δ_ω, times a linear term called proportional gain K_P. The proportional gain represents a drastic response of the controller, which, for a given error, aggressively turns the pitch angle in order to quickly cover the gap between the current and the rated shaft speed. The second term is equal to the integral of the error Δ_ω throughout the operation of the wind turbine, times another linear gain term, called the integral gain K_I. By monitoring the shaft speed error throughout the time integral, integral gain control is able to achieve mild pitch changes, when compared to proportional gain. These changes are needed to deal with the small, consistent in time, errors produced by the sudden pitch changes of proportional gain.

It is worth mentioning that PI controllers are not only met in pitch controllers. As it is later realized in Section 4.4, a PI controller can also be used for torque controlling. After all, the controller's basis is the error between the anticipated and the real shaft speed, which is also relevant for torque control.

2.4.4   Floating wind turbines control

The above-mentioned techniques demonstrate the fundamentals for controlling a fixed-bottom wind turbine. However, if the same variable speed and variable pitch (VSVR) controller is implemented in a fixed bottom and a floating wind turbine, without any additional modifications, it will introduce negative feedback (instability) to the system. The instability derives from the fact that, unlike the case of fixed bottom turbines, pitch response oscillations in floating wind turbines have a higher period than the controller's reaction time. This results in an aggressive controller.

The negative feedback is illustrated in the following simplified case of a floating wind turbine undergoing pitch motions, in a constant wind field. During the pitch oscillation, two different stages can be identified:

- When the nacelle moves downstream, the relative velocity felt by the rotor, V_{rel}, is less than the wind speed, V_w (Figure 11, left part). As a result, the controller pitches the blades following a lower wind speed scenario, causing a higher thrust force on the rotor (compared to the real wind velocity thrust force). Since the force does not point to the equilibrium position, the fact that it increases leads to additional destabilization of the floating wind turbine.

- In the opposite case, when the nacelle moves upstream, the relative velocity V_{rel} is greater than the wind speed V_w (Figure 11, right part). As a result, the controller pitches the blades for a case of higher wind speed, causing a lower thrust force on the rotor. In this case, the thrust force direction is to the equilibrium position, and due to its reduction, further instability is introduced.

Every time those two response stages are repeated, the difference between V_{rel} and V_w increases due to the additional work added to the system from the previous oscillation. Even though it is most likely that, at some point, the negative feedback is counteracted by the increasing damping-derived forces applied on the floater, this phenomenon increases the accumulated fatigue on the turbine (due to large oscillations)—moreover, the quality of the power output of the turbine is reduced.

Different techniques have been developed to address negative feedback. The simplest way is to modify (detune) the response time of the controller to make it slower than the pitch response. This approach is very common, and the way to do so is by introducing a time lag in the response of the controller in order to reach slower than the wind turbine pitch response. Another alternative is to filter out signals of the same frequency as the pitch response. However, this filtering method has to be very precise as, for every different environmental condition, the response frequency of the turbine differs.
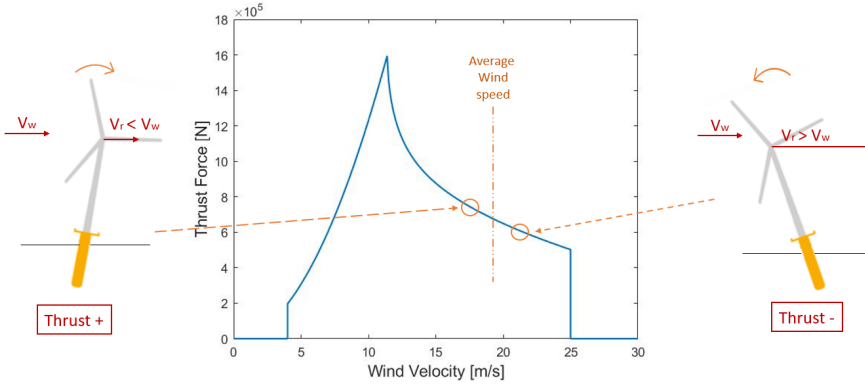
Figure 11: Negative feedback explanation

2.5 Design optimization

As stated by Kelley (Kelley, 2010 [15]), optimization is "the process of finding the most effective or favourable value or condition". Optimization processes are applied in all sciences. Especially in engineering fields that prioritise research and innovation, like floating offshore wind, optimization processes are getting increased attention due to the continued need for design refinement. In the following paragraphs, the main components of an optimization workflow are presented.

Essentially, for a given system, an optimization workflow manipulates specific variables defining the system (*design variables*), aiming to improve certain properties of the system (*objectives*). Simultaneously, the additional, non-optimized properties of the system should be maintained within specific limits (*constraints*). Mathematically, an optimization problem is defined as (Martins, 2021[16]):

$$
\begin{aligned}
\text{minimize} \quad & f(x) \\
\text{by varying} \quad & \underline{x}_i \leq x_i \leq \bar{x}_i \quad i = 1, \ldots, n_x \\
\text{subject to} \quad & g_j(x) \leq 0 \qquad j = 1, \ldots, n_g \\
& h_l(x) = 0 \qquad l = 1, \ldots, n_h
\end{aligned}
\tag{50}
$$

In this equation, $f(x)$ is the objective function, x_i is the design variable (can also be a set of design variables), and both $g_j(x)$ and $h_l(x)$ are constraint functions.

Starting from the *design variables*, x_i, those are the parameters that the optimization manipulates in order to improve the system objectives. Ideally, all the design variables affecting the optimization objectives should be included. However, increasing the variables leads to higher computational time (more designs must be produced and tested to converge to an optimum). The complexity of the way a parameter interacts with the design space increases, too. Therefore, it is important to identify those design variables with a stronger impact on the objective function, so that they can be included in the optimization problem formulation.

With the design parameters and the objectives defined, the next step is limiting the optimizer's design freedom based on the constraints of the problem. The constraint functions, $g_j(x)$ and $h_l(x)$, set the limits of the exploration space, within which the optimizer can produce possible alternative designs. This space is also known as the *feasible region* (Kelley, 2010 [15]). Constraints can either be geometrical or relevant to the behaviour of the structure. For instance, if the manufacturer of a wind turbine attempts to optimize the structure's power production, increasing the length of the blades is a promising option. However, if the length of the blades surpasses a specific limit, the blades will hit the ground. In this intuitive example, two key elements need to be addressed. Firstly, the constraints regulate the optimizer's design freedom. This is a very crucial consideration, as an over-constrained optimizer is not able to explore all the acceptable alternatives, one of which could be the real optimum one. On the contrary, an under-constrained optimizer exploits

design possibilities that do not serve the structure's purpose or make no physical sense. Secondly, these constraints need to be explained mathematically. If a particular design doesn't fulfil the constraints check, then the design variables are updated, and a new geometry is evaluated.

Finally, the *objective functions*, $f(x)$, evaluate the system's performance for a given set of design variables. The function inputs are the design variables, based on which it quantifies the objective value that a particular design holds. Depending on the optimization goal, different objective functions lead to different optimum solutions. In the wind turbine example, one way to increase power production is to increase the number of blades attached to the rotor. As long as the objective is increasing power production, this design is likely closer to the optimum. The same design, though, is likely to be less suitable (compared to the original) had the objective been the cost reduction.

The aforementioned components shape the optimization problem needed to be solved. What is missing from the workflow is the strategy that the workflow utilizes to obtain the optimum optimization outcome. The conditions under which a specific set of design variables will be considered the optimum are defined by the optimization algorithm (also called the optimization driver). Apart from defining the conditions for convergence to the optimum, the algorithm also specifies the designs tested in each iteration based on previous iterations. The fundamentals of optimization algorithms are given in Section 2.5.1.

There are many alternative criteria for terminating the optimization process. Some common termination criteria are based on the maximum number of iterations, the optimization time, and based on the convergence of the population. There are numerous indexes that can indicate convergence. For instance, in the pymoo optimization module [17] utilized in this thesis, such an index can be the objective space tolerance, f_{tol}, which terminates the optimization based on the change of the population's objective functions values.

Specifically for the case of optimizing a semi-submersible floating wind turbine substructure, a schematic overview of the optimization workflow is demonstrated in Figure 12. As shown, the optimization starts with an initial/reference design. This design is predetermined by the engineering team, and it can either be a preliminary design based on rough estimates of the structure's characteristics or one already proven to be suitable for the existing needs. For every design, the objective functions and constraint-related outputs are calculated. Most of these outputs (such as dynamic responses, power output, mooring tensions etc.) require simulating the substructure by utilizing a simulation model. With the required outputs computed, the constraints and the objective function are evaluated. Finally, if the design does not violate any constraint, the optimization algorithm checks if the design fulfils the criteria of the optimum one. If not, new design parameters are selected, and the next design is evaluated.
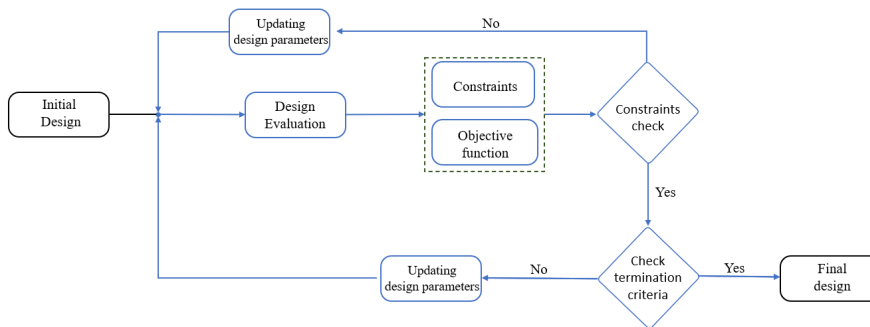
Figure 12: Design optimization process overview diagram

2.5.1   Optimization algorithms

There are numerous different optimization algorithms. However, they are separated into two fundamental categories: *gradient-based* and *gradient-free*.

As the name suggests, gradient-based methods optimize the design by evaluating if the objective function's minimum is reached based on its gradient. This process is demonstrated in Figure 13a. Gradient-based methods perform better when the number of variables increases (compared to gradient-free), even though deriving the function's gradient gets more complicated. However, this is also one of the drawbacks of such a method. Evaluating the gradient of complicated functions can introduce errors to the optimization algorithm. Moreover, the risk of minimizing to a local minimum and not the global one should be considered, as the method is dependent on the characteristics of the design space and the initial design chosen. Nevertheless, a common technique to overcome this issue is to initialise the optimization from different points of the design space.

In contrast with gradient-based optimization algorithms, gradient-free algorithms typically focus on investigating the characteristics of the design space to obtain the minimum (instead of investigating the initial point's gradient). Among gradient-free algorithms, it is very common that the initial approach consists of a collection of points in the design space. Based on the objective function's value for each point, and the optimization algorithm utilized, the optimizer evaluates the morphology of the design space and converges to the global minimum, as shown in Figure 13b. The big advantage of gradient-free methods is that they are simple to set up (compared to gradient-based). Also, they are more suitable when the optimization objective function is affected by more than one objective. However, gradient-free methods usually need more computational resources, compared to their gradient counterpart, due to the need to create designs throughout the whole feasible space. The most common ones used in offshore engineering applications are Genetic Algorithms (GA), Particle Swarm (PS) and the Nelder-Mead. In the upcoming Section 2.5.2, the Genetic Algorithm, which is used in this thesis, is further analyzed.
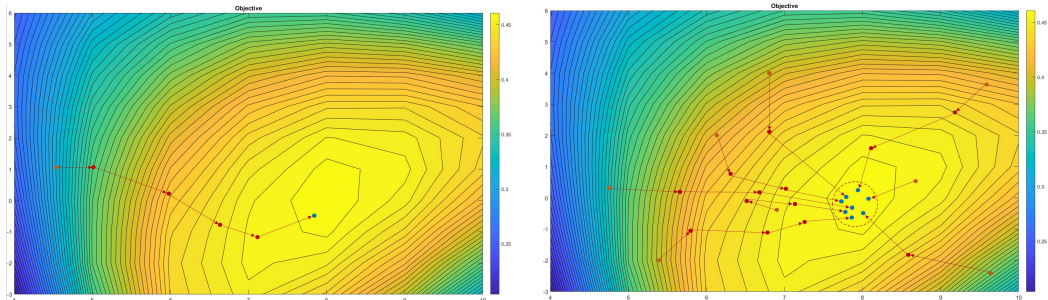
(a) Gradient-based optimization demonstration   (b) Gradient-free optimization demonstration

2.5.2   Genetic Algorithm

The *Genetic Algorithm* was introduced in the early 70s by John Holland. The idea behind this algorithm is based on the mechanics of natural selection. For a given population, the fitness of each individual is evaluated (by means of the objective function.), and the next generation of individuals (offsprings) is produced by combining the current population (parents), according to the fitness of each individual. For a design optimization case, such as the one studied in this thesis, the population consists of designs obtained from the combination of different design variables. The fitness of each individual is related to the property that needs to be optimized, such as the cost. More analytically, the workflow of the GA algorithm can be seen in Figure 14. The main steps of a GA algorithm are reported below, and represented in Figure 3 (Based on Kumar et al., 2010 [18]):

- The optimization is initialised with a specific number of individuals, that constitute the initial population. The population is produced randomly if not defined otherwise by the user.

- After the initial population has been introduced, each individual is evaluated based on their fitness for the optimization purposes. Fitness is rated based on specific functions, and in most cases, it is closely related to the objective function.

- In order to produce the next generation, a fragment of the existing population is selected for breeding the new one. The selection process is driven by the fitness of the individuals, evaluated in the previous step. However, a smaller percentage of individuals with a lower fitness value is selected for breeding. This ensures that the population maintains diversity, avoiding fast convergence to poor solutions.

- The selected population is used to produce the next generation of individuals. This process is called crossover, and it refers to the combination of parents into one or more offspring. Typically, a GA combines two parents to produce one new individual, as this combination is in line with biological natural selection. With that said, many GAs (especially more modern) produce offspring by more sophisticated parent combinations.

- After the crossover, a random mutation process takes place. This process introduces a mutation into the population, modifying the characteristics of the individuals to preserve some diversity in the population. After this step, the new population with higher average fitness value than the previous one is fully formed.

- With the exception of the initial population, all the other aforementioned steps are repeating inside a loop, gradually producing generations of greater (average) fitness. The algorithm is terminated according to the termination criteria specified by the user. The most typical two criteria for terminating the algorithm are either after a certain number of loops or by achieving a specific fitness to the population.

Apart from the original Genetic Algorithm, there are many alternative versions of this algorithm, like the Non-dominated Sorting Genetic Algorithm (NSGA-II), Biased Random Key Genetic Algorithm (BRKGA), Reference Point Based NSGA-II (R-NSGA-II), etc. In this thesis, the NSGA-II is utilized, and the algorithm is explained in Section 4.9.
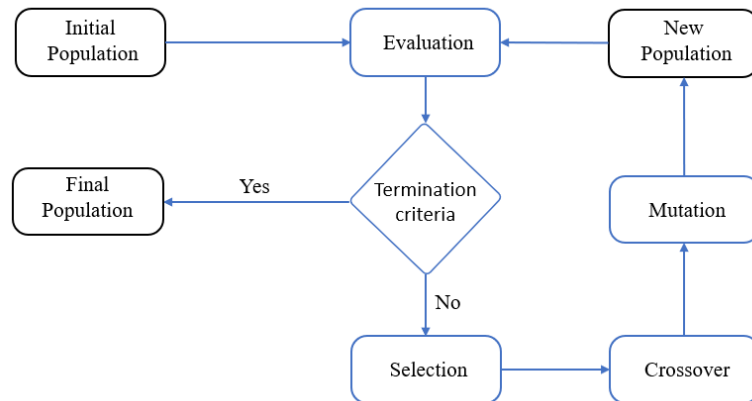
Figure 14: Genetic Algorithm workflow diagram.

2.5.3 Model analysis

A large portion of the computational effort involved in the optimization of a substructure for floating wind turbines is determined by the simulation of the substructure. As shown in Figure 12, the model simulation is connected with the constraints and objective function blocks. Depending on the constraints and objectives of the optimization problem, it is common that some of them are derived from a simulation model.

Floating wind turbines' analysis can be performed either in the Time Domain (TD) or in the

Frequency Domain (FD). In a time-domain approach, the structure's response is modelled, under a specific load condition, for a given timeline. Instead, in a frequency domain approach, the structure's steady-state response is simulated over a load condition for a wide frequency spectrum. The main differences between the two methods are also summarized in Table 1. Briefly, their main difference lies in the fact that, compared to FD analysis, a TD analysis is more accurate but also more computationally expensive.

For a given simulation, the modelling approach depends on the complexity and accuracy needed for the model. For the case of a substructure's optimization, frequency domain analysis is the one commonly used. This choice lies in the fact that, during an optimization process, a significant number of different designs are evaluated over a significant number of optimization iterations. What is more, if each substructure in a wind farm is optimized independently, then the computational time of a TD-based optimization can skyrocket compared to an FD-based optimization.

Nevertheless, it is not implied that optimizing a substructure does not require a TD simulation. In reality, a sophisticated way to optimize a substructure is to conduct an FD-based optimization at an initial stage. That way, areas of the design space with greater potential (for a specific objective function) can be identified without requiring an unrealistic computational time. At a later stage, the optimum designs should be further analyzed with a TD model in order to obtain more accurate results and investigate transient phenomena, which can be crucial for the integrity of the substructure. Even if, after the TD analysis, the (FD identified) optimum design ends up violating specific constraints, the design space used to find the optimum is shrunk, due to the identification of design space areas with low potential, according to the FD-based optimization results.

	Time domain	Frequency domain
Response	Transient steady-state	only steady-state
Effects	Linear non-linear	Linear
Accuracy	High accuracy	Low accuracy
Computational resources	High	Low
Simulated time	Specific - short	Infinite

Table 1: Frequency and Time domain simulations overview

2.6 Surrogate modelling

As already explained in Section 2.5.3, for the design optimization of a floating substructure, the simulation model requires significant computational resources to run. One of the most promising ideas to reduce the computational time is to replace the simulation analysis block with a simple model that predicts the analysis block's output for a given input. To make the prediction, this model should utilize a dataset of input/output pairs, based on which the model can be trained to make predictions. This model is called a *surrogate model*, and it is that it is a model able to approximate a function's output for a given set of inputs (S. Guo, 2020 [19]). The surrogate model operates dramatically faster, increasing the efficiency of the optimization, as the computationally expensive calculations are substituted with the model's predictions. Some commonly used models are the Polynomial Response Surface Models (PRS), support vector regression (SVR) model and the Gaussian process models (GP). More information about these models can be found in Jiang's book about surrogate modelling (Jiang, 2020 [20]).

Even though a data-driven surrogate model can indeed reduce computational time, implementation of the model requires specific steps to be followed. These steps are:

- Definition of the surrogate model's inputs and outputs. For a surrogate model replacing an analysis block, that means identifying the inputs needed by the analysis block and the outputs given by it.

- Utilizing an appropriate sampling strategy. This is a particularly important step, as the prediction capabilities of the surrogate model depend greatly on the quality of the sample.

- Producing the training dataset. With the dataset's samples known, each sample's set of inputs is analyzed (by the analysis block) in order to obtain its corresponding outputs.

- Fitting regression models to the dataset and evaluating which model is more capable of predicting the required outputs. This model is the most suitable surrogate model.

There are two main types of problems that surrogate models are applied: *classification* and *regression* problems. Classification problems involve the prediction of a categorical decision among a finite number of possibilities (i.e. 'yes', 'no', 'good', 'bad' etc.). On the contrary, in a regression problem, the predicted output is a continuous number (i.e. the cost of a structure). Additionally, regression problems can be split into single and multi-output regression problems. Single-regression problems involve the prediction of an output based on one input variable. Respectively, Multi-output regression problems deal with predicting a dependent variable based on multiple independent variables. Selecting the most suitable surrogate model depends on the inputs and outputs of the problem. Furthermore, it depends on the characteristics of the dataset, the relationship between inputs and outputs, and the dependency (if any) between the input or the output variables.

In this study, the performance of specific different surrogate models is evaluated for one multi-output and one single-output regression problem. These surrogate models are Linear Regression, Decisions Trees, Random Forests, Extreme Gradient Boosting, and Artificial Neural Networks. These models are expected to offer reliable predictions based on the problem characteristics. The following sections introduce the reader to the theoretical basis of these data-driven models.

2.6.1 Linear Regression

Linear Regression (LR) is a statistical process assuming that the relationship between the dependent (output) variable $y \in \mathbb{R}$ and N- dimensional input features (inputs) $\boldsymbol{x} \in \mathbb{R}^N$ is linear (Montgomery et al., 2012 [21]). In other words, the prediction function is given by:

$$y = \beta_0 + \sum_{i=0}^{N}(\beta_i \boldsymbol{x_i})$$ (51)

where β_0 is the intercept and β_i is the slope of $\boldsymbol{x_i}$. If the method predicts y depending only on one independent variable, it is a simple linear regression.

Even though the philosophy of linear regression is quite straightforward, the challenging part of this method is obtaining the best fit for a linear function, as, in most cases, a line is not able to pass through every output point. In order to obtain the best fit, the error between the line's predicted values and the real values must be minimized.

Even though there are numerous error evaluation variables, the most common one used in Linear Regression problems is the sum of squared errors (SSE), given by:

$$\epsilon = \sum_{i=0}^{N}(y_i - \hat{y_i})^2$$ (52)

where y_i is the real value and $\hat{y_i}$ is the value predicted by the surrogate model, determined by Equation 51.

The linear regression method is an easily interpreted method, as the only parameters affecting

the regression are the slope constants β_i. Furthermore, even compared to other surrogate models, linear regression is one of the most efficient prediction methods. Additionally, due to the linearity assumption, linear regression can be used to identify significant linear relations between input-output pairs. Even if a certain pair of input-output can be better predicted by another model, a significant accuracy by the LR model implies a significant linear relationship between the data. Nevertheless, non-linear relationships between input-output pairs cannot be captured by this model. Additionally, for multi-output regression problems, the method assumes independence between the y observations. Depending on the characteristics of the data, this assumption can significantly affect the performance of the method.

2.6.2 Decision trees

Decision Trees (DT) are named after the graphical representation of the algorithm's decision process, which resembles a tree. They are non-parametric models that can be used for both classification and regression problems. Even though a Decision Tree surrogate model is not utilized in this thesis, it sets the basis for the Random Forest model, which, together with the Extreme Gradient Boosting model, are the main models applied.

The basic idea behind a DT is that the input information passes through several evaluation levels, identifying specific traits of the input and leading to a decision on the output for such an input. Figure 15 demonstrates a Decision Tree. By observing the tree, three main types of elements can be distinguished (Song et al., 2015 [22]): a root node (blue-coloured), leaf nodes (green-coloured) and decision nodes (yellow-coloured, also called intermediate nodes). The root node is the starting point of the evaluation process, as there is no incoming line but only outgoing ones (indicating a splitting). This node is the starting point of the prediction, where the input of the problem is inserted. Based on the characteristics of the input, the root node assigns it to one of the available nodes. After splitting, the information can either be assigned to a leaf node or a decision node. If it is led to a decision node, the input data is re-evaluated on different criteria in order to be re-assigned. Regardless of the number of decision nodes that the input will pass through, in the end, it should end up at a leaf node, which corresponds to the given output, for the specific case.
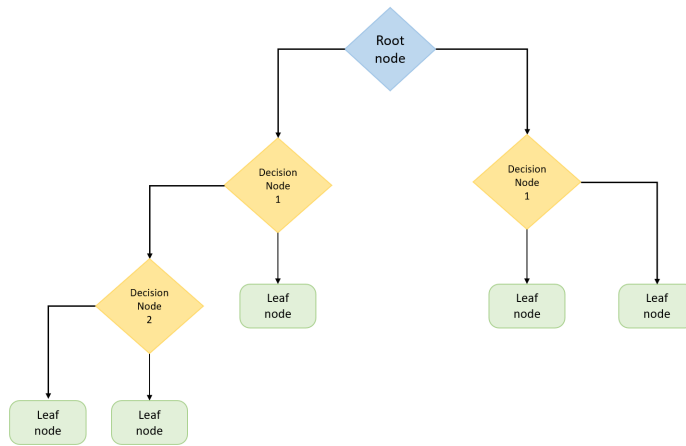
Figure 15: Decision trees workflow diagram. adapted from Son et al., 2015 [22]

It becomes clear that the philosophy of DTs is simple and intuitive. What is less intuitive, however, is how the algorithm structures the decision tree, as there are numerous alternative ways to structure a DT for the same problem. This process, named tree selection, is based on the concepts of entropy and information gain. In a decision process, entropy indicates the randomness (or impurity) of the samples processed and is given as (Shannon, 1948, [23]):

$$E = -\sum_{i=1}^{N} p_i \log_2 p_i \tag{53}$$

where p_i is the probability of selecting a sample that leads to the decision i. The higher the randomness of the system, the higher its entropy. Thus, if the samples of a system that lead to different outcomes are equally distributed, the entropy, E, of the system is maximized. Similarly, if all the samples result in the same decision, the system's entropy is zero.

Every time the samples are split, the system's entropy should be reduced, as there is less randomness in the new samples (considering that the initial samples were split according to some relevant criteria). This entropy reduction can be calculated based on the information gain, which is calculated as:

$$Gain = E_i - E_{i+1} \tag{54}$$

Where E_i if the system's entropy before splitting the samples and E_{i+1} is the total entropy of the split subsystems produced. By testing the information gain of alternative splits, the surrogate splits the samples into nodes that maximize the information gain.

Decision trees are a very popular supervised machine-learning method, mostly due to their simplicity and their wide range of applications, as the prediction process is not dependent on the statistical properties of the sample. Plotting the tree is also very easy, making model examination feasible. However, for a complicated problem with the need for many splits in the data sample, DTs are prone to overfitting the data. Additionally, they are considered a poor choice for data containing noise.

2.6.3 Random Forest

Random Forest (RF) is an ensemble learning method based on the concept of decision trees, initialized by Breiman (Breiman, 2001 [24]). Similarly to decision trees, RF can also be used for classification and regression problems. Essentially, the method predicts the output of a problem by utilising the predictions of more than one decision tree.

More analytically, Random Forests operate inspired by the "divide and conquer" principle (Biaou et al., 2016 [25]). According to it, the initial data sample is randomly divided into smaller subsamples. For every sub-sample, a Decision Tree grows. If the method predicts a Classification problem, the prediction is extracted following the majority vote (which classification was given by most DTs). On the other hand, if the method predicts a regression problem, the prediction of each DT is aggregated into a single one. The most common way the RF prediction is calculated in the regression case is by calculating the average of the DTs output.

Regarding the decision trees grown to represent each sub-sample, it is worth mentioning that, unlike the Decision Tree method, the tree selection is not always driven by maximizing the information gain. There are many different approaches to the formation of the tree. For instance, One way to increase the trees' randomness is to apply random split selection at each node (Dietterich, 1998 [26]), in which case the split criteria is selected among the K best splits.

Aside from holding the same advantages as decision trees, the application of Random Forests effectively tackles the weaknesses mentioned in the former. Random Forests can be implemented in a wide range of sample sizes, from very small to very big ones, by scaling up under the "divide and conquer" philosophy. Unlike DTs, the method is also very popular for its accuracy, even in sample spaces of high dimensionality. It is also less prone to noise or being biased by overfitting the sample, as there is not an individual tree making the predictions.

2.6.4 Extreme gradient boosting

Extreme gradient boosting (XGBoost) is another decision tree-based method, and even more specifically, it can be considered an enhanced random forest method. Similar to the RF method,

XGBoost predicts the outputs of a problem based on the prediction of multiple decision trees. What differentiates the XGBoost from RF is that, instead of growing decision trees randomly, this method estimates its prediction error for a given number of trees and selects to grow a new tree that can maximize the error reduction. This section reports the key concepts of XGBoost, based on Chen's publication (Chen, 2016 [27]).

For a given training dataset, the XGBoost algorithm begins by growing one single decision tree. For the decision tree produced (and for each one following this one), the algorithm evaluates the tree's prediction error and structural complexity. The tree's error is given by the Loss function $l(y_i, \hat{y}_i)$. This function can be defined by numerous error indicators, such as the Mean Square Error, given by:

$$l = \frac{1}{N} \sum_{i=1}^{N} (y_i - \hat{y}_i)^2 \tag{55}$$

where N represents the number of samples, y_i is the predicted output, and $\hat{y}_i$ is the real output of a given input, $\boldsymbol{x_i}$. The complexity function, Ω, penalizes decision trees with complicated structures. It is calculated as:

$$\Omega = \gamma \cdot \text{ T} + 0.5 \cdot \lambda \cdot \Sigma \left| \boldsymbol{w_i} \right|^p \tag{56}$$

where T corresponds to the number of leaf nodes, w_i is a weighing factor for each leaf node, and γ, λ and p are constant hyperparameters, regulating the way that complexity is penalized.

Based on the loss and complexity functions, the model's performance can be estimated by its objective function L, being:

$$L(\phi) = l\left(\hat{y}_i, y_i\right) + k\Omega\left(f_k\right) \tag{57}$$

Where f_k is the function of the kth Decision Tree, and k is the Tree index. With the objective function defined, the XGBoost algorithm aims to introduce new decision trees to the system in order to reduce the objective function value. With a new decision tree introduced to the system, and for n existing decision trees, Equation 57 becomes:

$$L^{(t)} = \sum_{i=1}^{n} l\left(y_i, \hat{y}_i^{(t-1)} + f_t\left(\boldsymbol{x_i}\right)\right) + \Omega\left(f_t\right) \tag{58}$$

where $f_t(x_i)$ corresponds to the function of the new t^{th} decision tree introduced to the system. This new tree function should be defined considering the existing objective in the system in order to contribute to the reduction of it.

There are many trees that can result in the objective function's reduction. In order to maximize the effect of every new decision tree, the algorithm utilizes the gradient of the objective function $\partial L / \partial y$. That way, the higher error prediction areas can be identified, in which the new decision tree is needed to reduce the error the most.

When a new decision is grown, the model's objective function (Equation 58) is re-evaluated, together with its gradient $\partial L / \partial y$, and a new decision tree grows according to the same process. Also, during the re-evaluation process, a weight factor is assigned to each decision tree. The above-mentioned process is repeated for every new Decision Tree introduced to the system. When the desired number of decision trees is reached, the algorithm predicts $\hat{y}_i$ based on the prediction of each Decision Tree. Mathematically, this is given by:

$$\hat{y}_i = \sum_{k=1}^{n} (f_k(\boldsymbol{x_i})) \tag{59}$$

As a surrogate model method, the Extreme Gradient Boosting method holds the same advantages as the Random Forest method. At the same time, the algorithm's orientations in growing decision trees in areas of lower prediction accuracy increase the performance of XGBoost, compared to Random Forest. Furthermore, due to the algorithm's tendency towards decision trees of lower complexity, the structure of the prediction model is clearer and easier to interpret compared to Random Forests. Additionally, even though the decision tree selection algorithm requires some additional steps, compared to the RF method, there is no noticeable computational resources increase compared to RF. Nevertheless, it is important to note that, regardless of the expected improvement in accuracy, the performance of a method depends significantly on the characteristics of the training dataset.

2.6.5 Artificial neural networks

Artificial Neural Networks (ANNs) are surrogate models mimicking the way biological neurons operate. Similarly to most of the aforementioned models, ANNs are surrogate models able to predict both regression and classification problems, and they are also very popular for their prediction capabilities in non-linear problems. The following theory is based on Bishop's book on surrogate models (Bishop, 2006 [28]).

ANNs consist of artificial neurons. The structure of a neural network and the way that each neuron operates is shown in Figure 16. According to it, a neural network is an arrangement of neurons in layers. Each neuron (also mentioned as a node) receives input signals from the neurons of the previous layer, x_i, and processes them in order to export specific output signals. These signals are assigned to specific neurons. For the final receiving neurons, the output signals produced are the model's outputs.
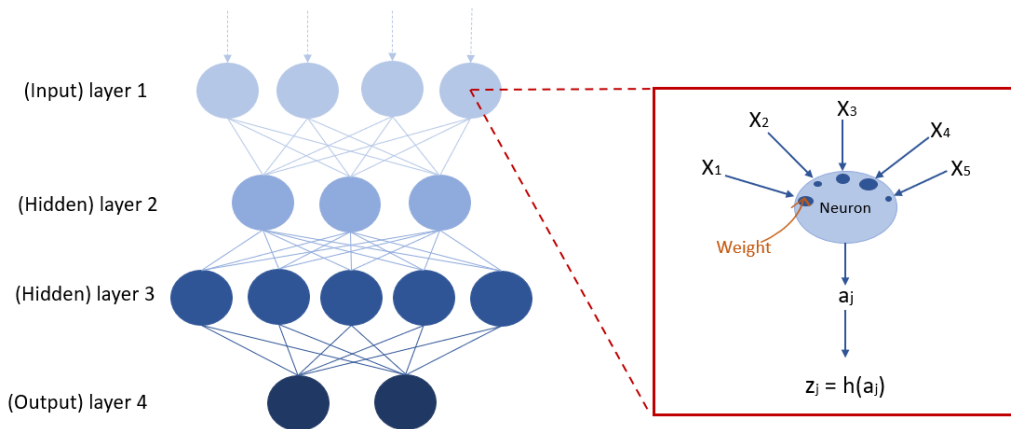
Figure 16: Artificial neural network diagram (left). Artificial neural neuron diagram (right).

For a given neuron in a layer l, the outputs assigned to the next layer's neurons are calculated in two steps. Firstly, the weighted sum of every input signal, a, is calculated. Assuming D inputs and M outputs, a is given by:

$$a_j = \sum_{i=1}^{D} w_{ji}^l x_i^l + w_{j0}^l \tag{60}$$

where j corresponds to the output index ($j = 1, ..., M$), i indicates the input index, w_{ji} is the weighting factor for each input-output set, and w_{j0} is a bias term. Additionally, with the exception of the first (input) layer, the input signal, x_i^l, is equivalent to the output signal, z_j^{l-1}, from the previous layer.

The weighted sum average term, a, is also called the *activation*. With a computed, each activation is translated into the output signal based on a differentiable, nonlinear *activation function*, $h(\cdot)$:

$$z_j = h(a_j) \tag{61}$$

There are many alternative activation functions. The most common ones utilized are sigmoid, hypertan or Rectified Linear Unit (ReLU) functions. Those functions are given as:

$$h(a_j) = \frac{1}{1 + e^{-a_j}} \text{ , for sigmoid} \tag{62}$$

$$h(a_j) = tanh(a_j) \text{ , for hypertan} \tag{63}$$

$$h(a_j) = max(0, a_j) \text{ , for ReLU} \tag{64}$$

With the output computed, each node of the l layer passes the output signal to the next layer nodes, $l + 1$. Respectively, for each node, the activation is given as:

$$a_k = \sum_{j=1}^{M} w_{kj}^{(l+1)} z_j + w_{k0}^{(l+1)} \tag{65}$$

where k is the output index. As shown, the input index of the $l + 1$ layer is j, which is the index used as the output index of the l layer.

It is important to mention that every node contained in the same layer of the ANN utilizes the same activation function. Training the surrogate model aims to obtain the most suitable w_{ij} combinations in order to achieve the best prediction.

3 Literature review

Many researchers have applied optimization techniques on the design of a floater, mainly in order to produce more economical and reliable designs. In most cases, the optimization involves a specific floater type. Among all the different ones, the most extensively examined is the spar-buoy. For instance, Leimeister et al. [29] examine the optimization of a 5 MW spar floating turbine. However, as wind turbines turn larger, so do floaters. In their more recent work, Hegseth et al. [30] design an optimization framework for the case of a spar-buoy floater able to support the DTU 10 MW reference wind turbine. Similarly, Pollini et al. [31] focus on the same type of substructure suitable for mounting the IEA 15 MW reference turbine.

Abdelmoteleb et al. [32] investigate a semi-submersible floater concept, aiming to optimize an upscaled design based on the VolturnUS-S platform [33], able to support a 25 MW wind turbine (assumptions are made regarding the turbine characteristics). Ferri et al. [34] optimize the design of a semi-submersible floater able to support a 10 MW wind turbine, after upscaling the OC4-5 MW platform (Robertson et al., 2014 [35]). The optimization objective in this research is to produce a design that reduces the dynamic responses of the floater. Finally, there is significant research on TLP models, such as the one by Lee et al. [36], focusing on the optimization of a TLP floater designed to support the 5-MW NREL reference wind turbine (Jonkman et al., 2009 [37]), by utilizing a Non-dominated sorting genetic algorithm-II (NSGA-II).

Nevertheless, there are cases of optimization frames offering more design freedom. The aim in these cases is to produce an optimizer able to exploit a variety of different concepts, finding the most suitable one for each case. Hall et al. [38] developed a *multi-objective optimizer*, able to explore the whole spectrum of spar-buoy, semi-submersible and TLP concepts. This is achieved by manipulating the main differences of these concepts, being the number of columns and the mooring line configurations.

Over the following paragraphs, the ways that many researchers approach the different optimization stages, described in Figure 12, are addressed.

3.1 Design variables

As already mentioned, the first step for setting up an optimization process is defining the design variables. Those are the parameters that the algorithm is allowed to optimize. Since this is a structural design problem, it is natural that the design variables optimized are, in the majority of cases, geometrically related. It is worth noting that, ideally, all design specifications would have to be included in the optimization parametric variables. However, such a decision would make a computationally expensive optimizer, as more designs would need to be analyzed, and also, it would complicate producing the new geometries due to the interaction between all the different variables. This is why it is always important to distinguish the parameters that can effectively affect the objective function output while also making reasonable assumptions for the non-optimized ones.

The design variables chosen are related to the substructure used. Even though in every research, there is a different set of design variables optimized, some variables are used in the majority of studies. For the case of a spar, as in the research from Leimeister et al. [39], the main design variables chosen are the draft, the diameter of the floater and the solid ballast filling height. Semi-submersible floaters are usually examined for design changes in the offset and the diameter of the outer columns, as done, for instance, by Renty et al. [40]. Finally, TLPs optimization typically revolves around the hull dimensions and the tendons size (Kim et al., 2016 [41]). To demonstrate more design variables, Table 2 indicates the aforementioned and some additional design variables, together with the floater type applied on and the publication reference.

Concept	Variable	Reference
Spar buoy	Draft	[31], [39], [42], [43]
Spar buoy	Floater diameter	[31], [42], [42], [43]
Spar buoy	Ballast filling height	[42], [43]
Spar buoy	Vertical fairlead position	[31], [42]
Spar buoy	Transition piece length	[31]
Spar buoy	Central column diameter	[31], [42]
Spar buoy	anchor radius	[31], [42], [43]
Semi-submersible	Outer column diameter	[38], [40], [44]
Semi-submersible	Outer columns offset	[38], [40], [44]
Semi-submersible	Pontoon width	[32]
Semi-submersible	Pontoon height	[32]
Semi-submersible	Heave plates radius	[38], [44]
TLP	Hull dimensions	[41]
TLP	Tendoms size	[41]
Spar/SS/TLP	Mooring line profile	[38], [44]

Table 2: Design parameters used for different floater types optimization.

3.2 Constraints

Choosing the right constraints is vital for realistic and effective design space exploration. Through the literature, there is a plethora of different constraints applied. For the purpose of this report, the constraints mentioned are distinguished into constraints related to the operation, the geometry and the mooring system of the floater.

3.2.1 Operational constraints

One of the most common constraints involves the acceptable pitch angle of the floater. This constraint ensures that the wind turbine will reach the anticipated performance and, more importantly, filters designs that could lead to floater over-turning. The most typical constraint for the pitch angle is a maximum sum of static and dynamic pitch angle of $10°$. Regarding the static pitch angle, Ghigo et al. [7] propose a simple calculation by only considering the thrust force (F_thrust) contribution on the platform, which is the most significant overturning force, divided by the pitch hydrodynamic stiffness, C_{55}, as seen in Equation 66:

$$\alpha^{static}_{\text{pitch, max}} = \frac{M_\text{thrust}}{C_{55}} \tag{66}$$

where M_thrust is the thrust force moment applied on the rotor, given by:

$$M_\text{thrust} = F_\text{thrust} \cdot d \tag{67}$$

with d being the distance between the center of the rotor and the center of buoyancy of the floater. C_{55} is defined as:

$$C_{55} = \rho_\text{w} g V \cdot \text{GM} \tag{68}$$

Here, ρ_w is the sea water density, g is the gravity acceleration, V represents the submerged volume and GM is the metacentric height. Equation 66 gives a good approximation of the static pitch angle but only accounts for the wind force. Abdelmoteleb et al. [32] introduce a more analytical approach to calculate static pitch, which also includes the moment applied from the mooring lines, and the weight of the RNA, due to its horizontal distance from the center of gravity, given by:

$$F_{55} = F_\text{thrust}\ h_\text{hub}\ - m_\text{RNA}\ x_\text{RNA}\ g - M_\text{moor} \tag{69}$$

with h_{hub} being the hub height, m_{RNA} the RNA mass, x_{RNA} the horizontal upwind distance between the RNA and the centre of mass and M_{moor} the moment applied by the mooring lines, for the allowable surge offset (18 m in this research). Due to the additional terms, the stiffness of the system in pitch is now calculated as:

$$C_{55} = \rho g I_{wp} + \rho \nabla g z_B - m_\text{tot}\ g z_G + k_m z_\text{fair}^2 \tag{70}$$

In this expression, I_{wp} is the second moment of area around the y-axis, ∇ is the displaced volume of water, z_B is the vertical position of the center of buoyancy, m_{tot} is the total mass of the floating system, z_G is the position of the system's center of gravity, and k_m is the equivalent stiffness of a horizontal spring located at the fairlead, which simulates the mooring line stiffness for the allowable surge displacement. Finally, this approach accounts for the coupling between surge and pitch, calculating the static pitch angle (η_5) as the solution to the system:

$$\begin{bmatrix} k_m & k_m z_\text{fair} \\ k_m z_\text{fair} & \rho g I_{wp} + \rho \nabla g z_B - m_\text{tot}\ g z_G + k_m z_\text{fair}^2 \end{bmatrix} \begin{bmatrix} \eta_1 \\ \eta_5 \end{bmatrix} = \begin{bmatrix} F_\text{thrust} \\ F_\text{thrust}\ h_\text{hub}\ - m_\text{RNA}\ x_{RNA} g - M_\text{moor} \end{bmatrix} \tag{71}$$

In this expression, C_{55} and F_{55} are the bottom right terms in each matrix, with the terms k_m and F_{thrust} representing the surge stiffness and excitation force, and $k_m z_{air}$ being the coupled stiffness terms.

Regarding the computation of the dynamic pitch angle, the most common practice is based on the platform's pitch standard deviation, $\sigma_{\xi 5}$, in a given sea state. Hall et al. calculate $\sigma_{\xi 5}$ as (Hall et al., 2013 [38]):

$$\sigma_i = \sqrt{\int_0^\infty |RAO_i(\omega)|^2\ S(\omega) d\omega} \tag{72}$$

where $S(\omega)$ is the incident wave's power spectral density and $RAO_i(\omega)$ is the Response Amplitude operator, in this case for the pitch motion.

Mathematically, the overall constraint applied by Hall et al. [38] is shown in Equation 73:

$$\bar{\eta}_5 + \sigma_{\eta_5} < 10° \tag{73}$$

In most cases, however, the maximum dynamic pitch angle constraint considers the maximum dynamic pitch angle to be equal with $3\sigma_{\eta_5}$, as done for instance by Pollini et al. [31].

Another crucial response is surge displacement. Although not harmful for the platform-turbine system, surge jeopardizes the integrity of the dynamic cable and especially when it extends further than certain limits. The best way to ensure the cable's safety is by constraining the accepted surge of the system. This limit is related to the site's depth. For a depth of 320 m, Pollini et al. [31] consider a maximum surge of 50 m, for optimizing a spar-buoy platform. In the same research, the minimum surge period is also constrained, at 80 sec. For the case of TLP platforms, however, the maximum acceptable surge displacement is lower due to the nature of their pre-tensioned mooring system, which raises concerns about the integrity of the system for significant horizontal displacements. For instance, Kim et al. [41] set a maximum surge limit of 14.0 m for survival conditions.

Regarding the turbine operation, a widely used constraint is about the fore-aft nacelle acceleration (α_{nac}). As a constraint, the accepted nacelle acceleration, α_{nac}, is typically limited to be less than $0.2g$ (Pollini et al., 2021 [31]). Another way to constraint the nacelle motions is based

on the standard deviation of α_{nac}, $\sigma_{a\ nac.}$, as this variables is closely connected to the fatigue experienced by the wind turbine. As a matter of fact, this variable is also implemented into the objective function for certain cases (Hall et al., 2013 [38]), because lower fatigue increases the lifetime of the wind turbine. This is further examined in Section 3.3. Calculating $\sigma_{a\ nac.}$ is based on Equation 72, with the difference that this time the nacelle RAO is calculated as:

$$RAO_{a\ nac.}(\omega) = -\omega^2 \left[RAO_1(\omega) + z_{\text{nac.}}\ RAO_5(\omega) \right] \tag{74}$$

which leads to Equation 75:

$$\sigma_{a\ nac.} = \sqrt{\int_0^\infty \left| RAO_{a\ nac.}(\omega) \right|^2 S(\omega) d\omega} \tag{75}$$

3.2.2 Geometrical constraints

Geometrical constraints are introduced in order to address two main issues: firstly, the size of the floater should be within some limits, as the dimensions of some designs can be constraint by manufacturing and installation-related factors. For instance, the available draft of specific harbours or the maximum column diameter that a manufacturer can produce constrains the dimensions range. Additionally, geometrical constraints block the optimizer from creating non-physical or irrelevant designs.

Such a case can be depicted in Figure 17, where (hypothetically) the objective is to optimize the design of the VolturnUS-S concept, ending with a floater of lower pitch response. One way to reduce heave is to increase the outer columns offset, increasing the pitch stiffness of the structure. Simultaneously, in order to preserve the buoyancy equilibrium, the pontoons should become thinner, leading to the bottom design on the right of Figure 17, for an unconstrained increase in the columns offset. Even though this design can indeed reduce pitch, it is not hard to prove that, after some length, structural concerns arise. Similarly, if the objective was to reduce the heave response, one way to achieve it is to increase the outer columns distance. In this case, the top result of the same figure results in a very expensive, impractical design.

Depending on the way the optimization method operates, different sets of constraints can be defined. Adbelmoteleb et al. [32] limit all hull dimensions to 120 m. Furthermore, Dou et al. [42] set a constraint about the buoyancy center, which should always be higher than the mass (for a spar platform).

Lastly, it is worth mentioning that, even the design variables are defined within certain bounds, as for the aforementioned manufacturing and installation related reasons, there are limits to the dimensions that they can reach.
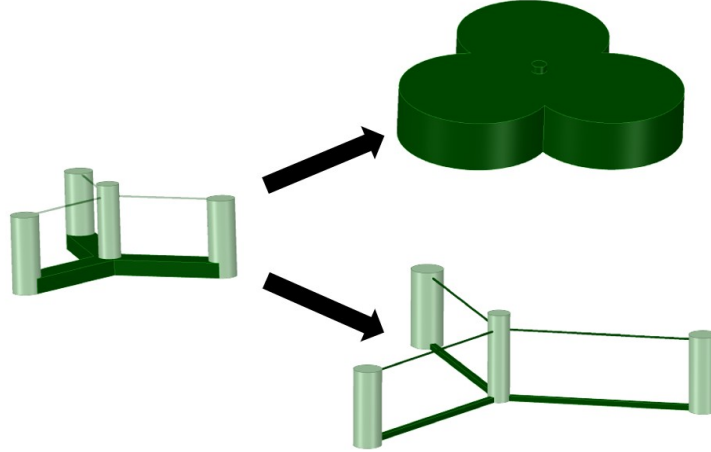
Figure 17: Example of non-constrained optimization outcomes

3.2.3 Mooring lines constraints

Due to the different types of mooring lines and configurations, there are lots of different constraints applied. Starting from the catenary mooring system, the constraints used handle the possible fully-suspended length of the line. On one hand, a catenary line should be long enough to actually connect the fairlead with the anchor point. What is more, the line is a catenary one, only if the length is greater than a straight line connecting the two points (see Figure 18), as given by Equation 76:

$$l > \sqrt{X^2 + H^2} \tag{76}$$

where X is the fairlead-anchor horizontal distance and H is the water depth.

On the other hand, there is a maximum limit to how long can a mooring line be in the water. This is explained by the fact that the mooring line carries its own suspended weight in water, which should not surpass the breaking limit of the line's chains. According to DNV-GL [13], the maximum load, Q, that a chain of specific diameter, d_c, can withstand is found after solving Equation 77:

$$\frac{Q}{w_{\text{water}}} = 103.84 \left(44 - 0.08 d_{\text{chain}} \right) \tag{77}$$

By knowing Q, the maximum chain length can be calculated based on:

$$\frac{S_{\text{max}}}{D} = \sqrt{\frac{2}{SF} \cdot \frac{Q}{w_{\text{water}} \, D} - 1} \tag{78}$$

In this expression, SF is a safety (typically equal to 0.95), D is the water depth and w_{water} is the effective submerged weight per meter of the mooring chain. This formula is also mentioned by [7].

Another phenomenon that needs to be controlled is the force applied to the anchor, as in reality vertical forces are to be avoided at the anchor. For this purpose, Pollini et al. [31] set a maximum suspension limit of 75%. Another simple approach is to set an upper vertical force limit at the end of the mooring line, based on the anchor used.

In the scenario of a taut-leg mooring system, there are some different constraints to consider. Starting from the geometrical details, the taut line (original) length should be equal or less than a

straight line between the fairlead and the anchor. This geometrical constraint is demonstrated in Figure 18, together with the corresponding one from the catenary mooring case.

Apart from constraining the taught line geometry, its behaviour should also be limited. The main concern about taut-leg mooring lines is the experience of snap-loads. Snap loads appear when a taut line becomes slack, and then regains its tension (Hall et al. [38]). A criteria to avoid this occurrence is the one shown in Equation 79:

$$\bar{T}_{\text{line}} - 3\sigma_{T_{\text{line}}} > 0 \tag{79}$$

in this equation, $\bar{T}_{line}$ is the steady-state line tension and $\sigma_{T_{line}}$ is the RMS line tension maximum variation. This constraint is very practical, as it includes the geometrical constraint mentioned above, because $\bar{T}_{line}$ is very small, if the line is not stretched.
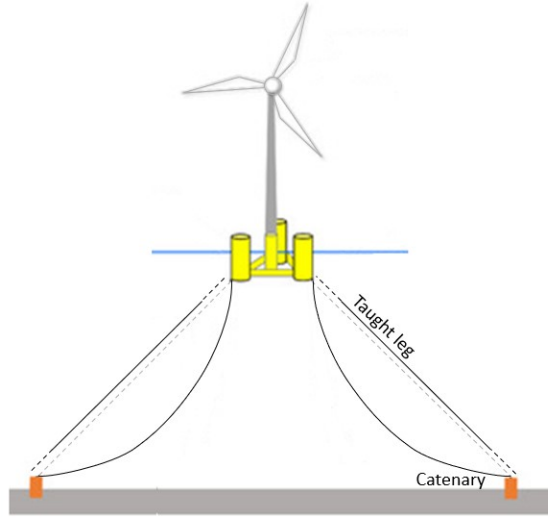
Figure 18: Geometry difference between catenary and taught leg mooring system

3.2.4 Additional constraints

Depending on the case, there are also additional constraints included, serving specific purposes. One example is the definition of a maximum platform cost. Even though cost is almost always part of the objective function, if the function is multi-objective one, then setting a maximum limit for the amount of material used (hence, the cost of the floater), can filter designs that need an abundance of material, without simulating their response. Such a constraint is applied by Hall et al. [38], with a maximum floater cost constraint of 9 million euros.

Frequently, constraints are set regarding the natural frequencies of the floater. For all degrees of freedom, their natural frequencies should be away from ranges of high power concentration in the wave spectrum. Although those areas depend on the specific environmental conditions of the installation site, a general wave high power spectrum is between 0.04 and 0.2 Hz (Collu et al., 2016 [45]). Additionally, the floater's DOF and the tower's bending natural frequency should be away from the rotor rotation and passing blade frequencies, 1P and 3P. Regarding the natural frequencies of the floaters, in most cases, they are outside of those areas. As a reference, Table 3 mentions the typical natural frequencies of the three main types of floating structures (T. Moan et al., 2018 [46]). A natural frequency constraint can be seen by Abdelmoteleb et al. [32], limiting the natural periods of the rigid floater to be higher than 20 sec (0.05 Hz).

The structural integrity of certain members can be ensured by constraints. These constraints are mainly applied to slender connection elements, such as braces on a semi-submersible, as there are

	Spar	SemiSub	TLP
Surge/Sway	0.005-0.025	0.008-0.25	0.015-0.05
Heave	0.02-0.05	0.025-0.07	0.2-2
Pitch/Roll	0.02-0.04	0.02-0.04	0.2-1
Yaw	0.025-0.2	0.0125-0.02	0.03-0.2

Table 3: Typical frequencies (Hz) of floating substructure concepts

many different kind of failures that can take place, like buckling. One example is the constraint used by Hall et al [44]. What is also different about this example, is that the constraint is not used to filter designs, but to define one design parameter. More analytically, for the geometry and the load case illustrated in Figure 19, the maximum force considered to applied on the braces is:

$$P_{\text{des}} = \frac{\max\left(\rho\forall_O g, T_{\text{Line max}}\right)}{\sin(\theta)} \tag{80}$$

where the maximum compressive load, P, is either equal to the buoyancy force created by the outer column ($\rho\forall_O$, or the maximum steady-state vertical load, applied from the mooring chain. θ refers to the angle of the member, with respect to the horizontal plane. This maximum load should be less than the critical buckling load, calculated as:

$$P_{\text{Crit}} = \frac{\pi^2 \text{EI}}{L^2} \tag{81}$$

As L, the length of the diagonal brace is chosen, as this longer brace is the most crucial one to fail. Finally, If $P_{crit} \leq P$, the diameter of the member can be computed from Equation 82:

$$D = \left(\frac{8}{\pi^3} \frac{P_{\text{Crit}} L^2}{Ek}\right)^{1/4} \tag{82}$$
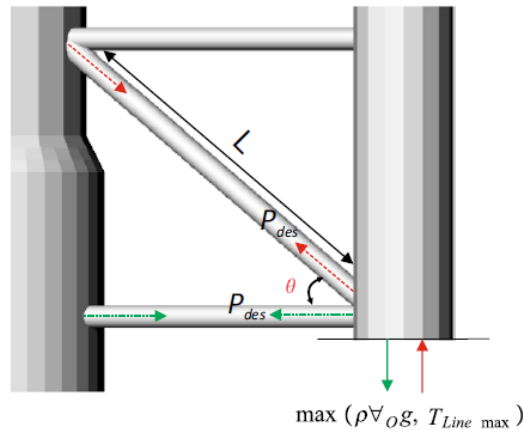
Figure 19: Loads demonstration for a diagonal truss member (Hall, 2018 [44])

3.3 Objective function

In the majority of cases, the optimization goal is to create a design that fulfills the constraint criteria, while reducing the cost. Thus, the most widely used objective function is the cost function.

While easily explained, the main problem with applying such a function is that material cost changes over time. Especially over the past years, steel price has skyrocketed and it can be seen

changing even on a monthly basis. However, this issue doesn't stop the objective function for operating effectively. That is because, even if the prices change, the driving cost parameter is the mass of structural elements used. That said, even if the objective function reports an incorrect cost, the reduction succeeded is real. However, This claim's credibility is set into question when the floater cost is obtained by more than one materials, as in this case different design changes affect the cost reduction differently, based on the material reduced. Still, the cost ratios between different materials is relatively stable. In the end, even if the materials cost changes, it is always possible for a user familiarized with the optimization algorithm to update those values.

Based on the literature, the cost function mainly includes the mass of steel and the mooring lines cost, as done by Karimi et al. [44]:

$$C(x) = C_{\text{Platform}}(x) + C_{\text{Mooring}}(x) + C_{\text{Anchor}}(x) \tag{83}$$

In Equation 83, $C_{\text{Platform}}(x)$ represents the platform cost, given by the steel mass multiplied by the cost of steel (2.5 \$/kg). $C_{\text{Mooring}}(x)$ and $C_{\text{Anchor}}(x)$ account for the mooring system cost, the former of which is given by multiplying the total mooring line length with the corresponding cost per meter and per kg (in order for different chain diameters/weights to be examined). The price of the anchors is simply equal to the individual price times the number of anchors. In this research, the price of ballasts is not included, due to the low cost of it, when compared to steel.

Reducing the amount of materials needed is, indeed, a very effective way to succeed the optimization goal of a lowering the project cost. An alternative, commonly used variable when calculating the cost of an energy-related investment, is the Levelized Cost of Energy (LCOE, Johnston et al., 2020 [47]):

$$LCOE = \frac{\sum_{t=1}^{n} \frac{I_t + M_t + F_t}{(1+r)^t}}{\sum_{t=1}^{n} \frac{E_t}{(1+r)^t}} \tag{84}$$

Briefly, LCOE describes the cost of a project for each MW produced (\$/MW. This is a convenient way to understand how profitable is an investment. If the electricity selling price (also \$/MW) is higher than the LCOE, then the project (under the assumptions made) is profitable. One of the key components of LCOE is the lifetime of the project (mentioned as t). It is natural that, if the lifetime of the project is prolonged, the investment cost will be distributed further.

To present the cost of the investment more accurately, many researchers take a more realistic approach to optimize the platform cost. Apart from reducing the material mass, there is also a need to obtain a design that will experience less fatigue, hence longer lifetime. An application of this idea can be found in the research conducted by Hall et al. [38], where the fatigue of the structure is assumed to be closely related to the nacelle acceleration. Thus, a *multi-objective* function is initiated, combining the nacelle acceleration parameter, based on Equation 75, and the structure's cost. The objective function final form is:

$$\min J = W_1 \sum_{i=1}^{n} w_i (\sigma_{anac.})_i + W_2 \, \text{Cost} \tag{85}$$

where w_i is a weighting function, dependent on the different sea states, so that the sum of all the w_i is equal to one. Moreover, W_1 and W_2 are used to represent the importance of the 2 parameters (acceleration/fatigue and cost).

In another attempt to make a more realistic cost-related objective function, Hegseth et al. [30] include a parameter determining the quality of the power produced, which is related to the weighted average of the rotor speed standard deviation, given by:

$$\sigma_{(\dot{\phi})} = \sum_{i=1}^{N_{\mathrm{EC}}} p_i \sigma_{(\dot{\phi}),i} \tag{86}$$

where $\sigma_{(\phi,i)}$ is the the rotor speed standard deviation for condition i. Again, the two objective don't have the same weight. This is why, two weight constants W_1 and W_2 are defined, leading to Equation 87:

$$f = w_1 C_{\mathrm{total}} + w_2 \sigma_{(\dot{\phi})} \tag{87}$$

3.4 Optimization method

Different researchers use different optimization methods depending on the characteristics of the design space. However, between the two types of methods mentioned in Section 2.5.1, gradient-based optimization is the choice in the majority of cases. Although the reason of choice is not mentioned, it is expected that, since the kind of research that uses gradient-based optimization deals with a single floater type, the design space does not consist of local maxima/minima. Moreover, even if there is no strong argument to choose gradient-based against gradient-free regarding the design space, the additional number of simulations needed from a gradient-free optimization to sample the design space is more time-consuming.

Nonetheless, there are instances where gradient-free optimization is used. Such an application can be seen in [38], where a genetic algorithm (GA) is used. The philosophy behind a genetic algorithm is based on the evolution of species principle. Many random designs are produced that cover the design space, and, based on their performance, the better ones are kept, based on which new designs are made. In this research case, the design space is defined by a variety of different concepts, and it is expected that this is the reason why sampling the design space is a better approach.

3.5 Model analysis

Simulating the substructure behaviour occupies the majority of the optimization time. It is important then to use a high-fidelity model, able to simulate the real-world phenomena observed accurately, but also a computationally efficient one, in order to obtain reliable results within a reasonable time. Due to that, almost every research mentioned is based on frequency domain simulations, mainly due to the lower computational resources needed when compared to time domain simulations. In their research, Pollini et al. [31] simulate the floating wind turbine using QuLAF (Quick Load Analysis of Floating wind turbines), a frequency-domain linear model developed to provide great computational efficiency (Pegalajar-Jurado et al., 2018 [48]). Hall et al. [49] have also developed an open-source frequency-domain analysis model, called RAFT. There are also different model approaches depending on the efficiency and accuracy needed. Adbelmoteleb et al. [32] introduce a simple 2D model for calculating the dynamics of a floating wind turbine which, even though simple, reduces the computational resources needed to such an extend that there is no need to limit the number of design variables including, neither the size of the design space.

3.6 Surrogate model applications

Surrogate modelling applications for optimizing the design of a floater are limited. This research gap is also acknowledged by the research community, as stated by Ojo [50] in his recent publication. Most of the publications made in this field are quite recent. Faraggiana et al. [43] optimize a floater based on the OC3 spar buoy concept utilizing a GA algorithm in combination with the Kriging

surrogate model. The results indicate the potential of surrogate model-based optimizations to obtain reliable optimization results while increasing the solver's efficiency. In another study, Pillai et al. [51] optimize the mooring system of a floating wind turbine floater based on a GA algorithm combined with surrogate modelling. The surrogate model is trained based on validated simulation results. The research outcome is an efficient optimization framework, able to evaluate the trade-off between cost and fatigue optimization. Finally, in his research, Hassani [52] optimizes the design of the VolturnUS-S floater concept, by utilizing a GA algorithm combined with different surrogate models. The objective of this research was to optimize both cost and accumulated fatigue during the operational lifetime of the floater.

3.7 Thesis research questions

In the current literature on floating wind turbine platforms optimization, some clear research gaps can be identified. Firstly, the majority of these optimizations are single-objective ones. Also, there are limited cases of objective functions that account for fatigue or power quality. Especially for power quality, literature is limited. Secondly, the computational resources required by the optimization workflow to compute the optimum design have been identified by numerous publications. This observation highlights the need for more efficient optimization workflows.

Considering the research gap, this thesis investigates the trade-off between cost and power quality optimization of a floating wind turbine substructure. In parallel with this task, making the optimization workflow more efficient is investigated. The approach adopted is to substitute the computationally expensive sub-processes of the optimization algorithm with a surrogate model able to predict the output of the sub-processes. For that purpose, two alternative optimization workflows have been developed. Based on this research, the following research topics are investigated:

- After running the optimization workflow, what are the trade-offs between cost and power quality for the investigated substructure?

- What are the characteristics of the optimized designs?

- Compared to the original optimization workflow, how reliable are the surrogate-based workflows?

- Compared to the original optimization workflow, what is the efficiency achieved by the surrogate-based approach?

- What are the advantages and disadvantages of the two alternative approaches? What are the strong points of each approach?

4 Methodology

This thesis investigates the trade-off between cost and power quality optimization for a semi-submersible floater. In parallel, the potential of increasing the optimization efficiency by implementing a surrogate model is explored. In order to accomplish this research, the following actions are taken:

- Defining a reference design and design parameters. That reference model defines the basis on which every design modification is applied. Additionally, the reference environmental conditions are defined.

- Developing an algorithm able to import the reference geometry, calculate and apply the design changes resulting from the modified parameters, and export the new floater design in RAFT-compatible format.

- Definition of the design space, with respect to the design variables and constraints chosen.

- Training a surrogate model, able to replace the computationally expensive part of the optimization. In order to train the model, a suitable dataset should be produced. The accuracy of the model should also be examined.

- Developing the multi-objective optimization workflow. This requires defining the design parameters, constraints, objective functions and the optimization algorithm of the problem.

- Implementing the surrogate model, substituting the computationally expensive analysis blocks.

In the following sections, these actions are analyzed thoroughly.

4.1 Reference design

In this thesis, the reference floater used is designed based on the VolturnUS-S concept platform [33]. The reason for choice depends on two major factors. Firstly, the VolturnUS-S is a Semi-Submersible (SS) concept, containing a wide variety of design details, giving a lot of freedom for potential design configurations. Secondly, the platform is designed in order to support the 15 MW IEA reference turbine [14], a turbine of size and characteristics that comply with the modern wind energy industry's standards.

Even though the VolturnUS-S is a very suitable platform, the reference design used in this research is an alternative version of it, with the rectangular pontoons replaced by cylindrical ones with the same cross-sectional area. This modification is adopted to allow the application of the potential flow theory to the horizontal pontoons. In its current version, RAFT cannot apply potential flow theory on non-cylindrical members. Figure 20 shows the VolturnUS-S pitch added mass, as calculated in RAFT. Due to RAFT's limitation on rectangular members, the pontoon's added mass is calculated based on strip theory (further analyzed in Section 4.3). Compared to the added mass in pitch of the VolturnUS-S report [33], RAFT significantly underestimates the added mass by approximately 40%. For the purposes of this thesis, this underestimation error is considered significant. Thus, rectangular members are not analyzed in this thesis.
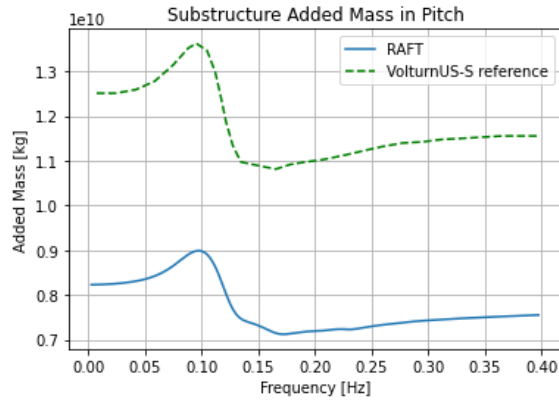
Figure 20: Added mass in pitch comparison between RAFT and the reference VolturnUS-S report[33]

Figure 21 illustrates the design of the reference platform. As the Figure shows, the floater consists of a central, inner column, and 3 symmetrically positioned (one every 120°) outer columns. Every outer column connects with the central one with a cylindrical pontoon, located at the bottom of the structure. The top of the vertical columns is also connected via thin cylindrical braces. Each column is also connected with a catenary mooring line at the fairlead point. Stabilization of the substructure is achieved by filling the pontoons with seawater and the outer columns with solid ballasts. The dimensions of the parameters mentioned are listed in Table 4. The wind turbine mounted on the floater is the IEA 15 MW, which is a reference turbine model, designed as a collaboration of the National Renewable Energy Laboratory (NREL) and the Technical University of Denmark (DTU), the design specifications of which can be found in Appendix A. More detailed sketches of the reference platform can also be found in Appendix B.
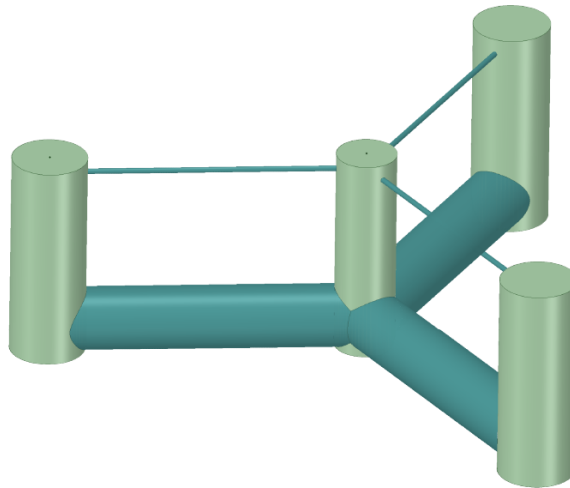
Figure 21: thesis research reference floater design.

Reference Substructure	
Outer column diameter (D_o)	12.5 m
Inner column diameter (D_i)	10 m
Inner-outer columns offset (L_o)	51.75 m
Pontoon diameter (D_{pon})	10.56 m
Draft height (T_d)	20 m
Mooring lines length (S_t)	840 m
Mooring lines thickness (d_c)	0.15 m
Fairlead height (w.r.t. sea level)	-14 m
Anchor-fairlead distance (X_{tot})	770.6 m

Table 4: Most important dimensions of the reference design.

4.1.1 VolturnUS-S comparison

Although there is no reference to the floater design used for this research, since it is based on the VolturnUS-S, it is worth comparing the two platforms' characteristics, as there are some similarities between the two models. More importantly, the factors leading to the differences between the two designs can be identified.

The first validation step involves the comparisons of the geometry-derived variables. Table 5 contains the mass-related data for the two cases. As mentioned in Section 4.1, the alternative design pontoons should have the same cross-sectional area as the originals. Since the rest of the components keep the same geometry, the almost identical hull displacement volume and water ballast mass indicate that the target geometry is achieved. There is one drawback, however, with mimicking the pontoons area, being the underestimation of the pontoons' steel mass. The shell steel mass is proportional to the pontoon outer area and calculated as:

$$M_{steel} = \sum_{member} A_{member} \cdot t_{wall} \cdot d_{steel} \qquad (88)$$

Based on that, and considering that the rectangular and cylindrical pontoons have a cross-sectional perimeter of 39 and 33.15 m, respectively, this leads to a steel mass difference of almost 3 t/m. This is why, steel mass in Table 5 is reduced by 276 t, compared to the VolturnUS-S. As a consequence, buoyancy equilibrium is achieved by increasing the solid ballast mass.

	VolturnUS-S	Alternative design
Hull displacement [m^3]	20 206	20 210
Solid ballasts [t]	2 540	2 742
Water ballast [t]	11 313	11 341
Steel mass [t]	3 922	3 646
Mooring chain density [kg/m]	685	684.5
Total mass [t]	20 093	20 067

Table 5: Comparison between the original VolturnUS-S design and the reference one's characteristics.

For the hydrodynamic properties of the two designs, their added mass and hydrodynamic stiffness are compared. Table 6 compares the hydrodynamic stiffness between the two structures. As shown, the stiffness calculated in heave is almost identical to the reference. This is expected, as heave stiffness only depends on the waterplane area of the semi-submerged columns, which is identical to the VolturnUS-S. However, pitch and roll stiffness is slightly increased. Considering the additional solid ballast mass in the alternative design, in order to balance out the reduction in water ballast, the alternative design's center of gravity is lowered. This results in a higher hydrodynamic stiffness according to Equation 99.

	Surge	Sway	Heave	Pitch	Roll	Yaw
VolturnUS-S	0	0	4.47 E+06	2.19 E+06	2.19 E+06	0
Alternative	0	0	4.48 E+06	2.48 E+06	2.48 E+06	0

Table 6: VolturnUS-S and alternative design stiffness comparison.

Regarding the added mass of the reference design, the surge and yaw added mass is higher than the VolturnUS-S ones, whereas the values for heave and pitch are lower. This is the result of the new pontoon members, as with a diameter of 10.56 m, the pontoon interacts with less added water volume in the heave/pitch direction. On the contrary, when moving along the x or y-axis, the 10.56 m pontoon diameter is higher than the VolturnUS-S 7 m of height, leading to a higher added mass. The graphs comparing these added masses can be found in Appendix C.

4.2 Reference environmental conditions

Even though the workflow described in this thesis can be used for any environmental case, a reference set of environmental cases is defined in order to set a basis for the production of the results. In order to define a realistic environment, the reference cases are based on the Buchan Deep Metocean report, the location where Hywind Scotland is installed, published by Statoil [53].

Eleven operational and one extreme environmental cases are defined. The operational cases begin by dividing the operational wind range of the IEA 15 MW (3-25 m/s) into 11 equal ranges (no wind-wave misalignment is considered). The probability occurrence of each wind speed is reported in Table 7.

Wind Speed [m/s]	4	6	8	10	12	14	16	18	20	22	24
Probability [%]	7.63	10.97	13.79	15.05	14.72	12.70	9.43	5.95	3.74	1.93	0.86

Table 7: Wind speeds and corresponding probabilities of occurrence (Buchan Deep report [53]).

With the wind speed amplitudes defined, the average Jonswap significant wave height, H_s, is given by Equation 89:

$$H_s = 0.719e^{0.0832V_w} \tag{89}$$

Equation 89 estimates the significant wave height by fitting a curve to the site's observed wind-wave characteristics. The comparison plot between the real and the estimated data is given in Figure 22. As shown, the fitted curve gives a reliable estimation of the real wave heights monitored.
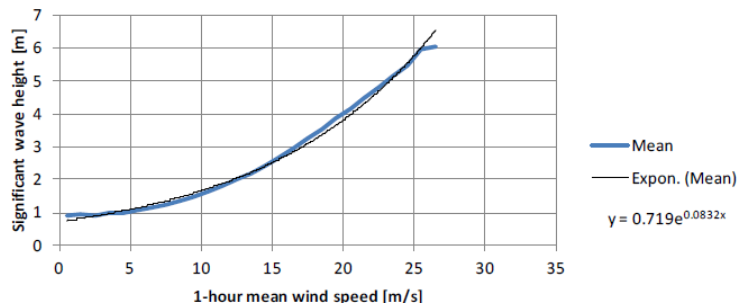
Figure 22: Comparison between the site observed (blue line) and theoretically calculated (black line) significant wave height H_s.

With H_s known, the corresponding Jonswap spectral wave peak period is calculated by linear interpolation between H_s and T_p, based on the observations given in Table 8.

Significant Wave Height - Hs [m]	1	2	3	4	5	6	7	8	9
Mean Peak Wave Period - Tp [sec]	6.8	7.9	8.8	9.6	10.4	11.2	11.9	12.6	13.3

Table 8: Corresponding mean wave peak period T_p, for a given significant wave height H_s (Buchan Deep report [53]).

The only environmental variable left to define for the operational conditions is turbulence, which is calculated as:

$$I_u(z) = 0.06 \cdot \left[1 + 0.43 \cdot \frac{U_0}{U_{ref}}\right] \cdot \left(\frac{z}{z_r}\right)^{-0.22} \quad \text{for} \quad U_0 \geq 12.0 \text{ m/s m/s} \tag{90}$$

and

$$I_U(z) = \frac{a \exp[0.09 \cdot U(z)]}{U(z)} \quad \text{for} \quad U_0 < 12.0 \text{ m/s} \tag{91}$$

where $U_{ref} = 10 m/s$ is the reference wind speed at a 10 m reference elevation Z_r, U_0 is the environmental case's wind speed, and z is the elevation variable (hub height in this case). $U(z)$ is the average wind speed for a given elevation, being:

$$U(z) = U_0 \cdot \left[1 + C \cdot \ln\left(\frac{z}{z_r}\right)\right] \tag{92}$$

where

$$C = 5.73 \cdot 10^{-2} \cdot \left[1 + 1.5 \cdot \frac{U_0}{U_{ref}}\right]^{\frac{1}{2}} \tag{93}$$

Finally, α in Equation 91 given by Equation 94:

$$a = I_U(z) \exp[-0.09 \cdot U(z)] \cdot U(z) \quad \text{for} \quad U_0 = 12.0 \text{ m/s} \tag{94}$$

Equation 90 to Equation 94 result in the TI shown in Figure 23. As seen from the graph, for most of the operational wind speeds, the calculated value is almost identical to the turbulence measured in the area.
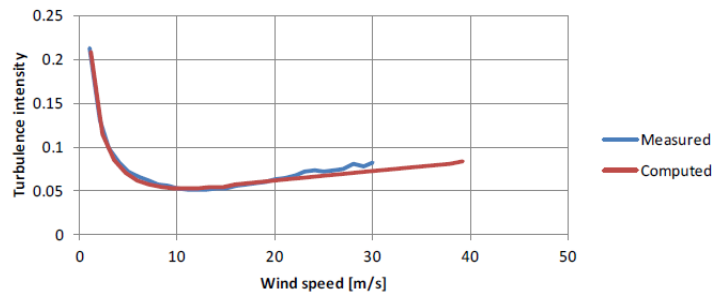
Figure 23: Computed and real turbulence intensity at an elevation of 100 m, for the location where Hywind Scotland is installed (Buchan Deep metocean report [53]).

The extreme environmental case is defined in order to examine the ability of the turbine to operate within the desired limits, even in the most (predicted) extreme environment. For that, a conservative approach is taken into account, which combines the rated wind speed (giving the maximum thrust force) with the location's 50-year return period wave, which is a wave of $H_s = 10.5$ m

and $T_p = 12.5$ sec (Buchan Deep metocean report [53]). Finally, since extreme turbulence is not included in the report, it is calculated according to the Extreme Turbulence Model (ETM), as defined in the IEC-61400 1 standard [54]. According to it, the extreme longitudinal wind velocity standard deviation, σ_1, is calculated as:

$$\sigma_1 = c \quad I_{\text{ref}} \left(0,072 \left(\frac{V_{\text{ave}}}{c} + 3 \right) \left(\frac{V_{\text{hub}}}{c} - 4 \right) + 10 \right) \tag{95}$$

In Equation 95, c is a constant equal to 2, V_{hub} is the wind speed at hub height, given by the power law:

$$\frac{V_a}{V_b} = (\frac{Z_a}{Z_b})^{\frac{1}{7}} \tag{96}$$

Finally, for the IEA 15 MW, which is a class IB wind turbine, I_{ref} is taken as 0.14 and V_{ave} is calculated as:

$$V_{ave} = 0.2V_{ref} \tag{97}$$

where $V_{ref} = 50$ m/s, according to the wind turbine's class.

Following the above-mentioned guidelines, the 12 environmental cases shown in Table 9 are formed. As shown in the table, every environmental condition is shown, with its corresponding probability of occurrence. The only exception to this case is the probability of the extreme environmental case, which is zero. The reason for that is that the extreme environmental case is the most harsh environmental condition that the wind turbine should withstand. However, the probability of occurrence of such a case is not expected to contribute to the power produced from the turbine.

Wind Speed [m/s]	T.I. [%]	Wave Height [m]	Wave Period [sec]	Probability [%]
4	24	1.00	6.80	7.63
6	20	1.18	7.00	10.97
8	17	1.52	7.37	13.79
10.5	16	1.72	7.59	15.05
12.5	15	2.03	7.93	14.72
14	14	2.31	8.17	12.70
16	14	2.72	8.55	9.43
18	14	3.22	8.97	5.95
20	13	3.81	9.44	3.74
22	13	4.48	9.99	1.93
24	13	5.30	10.64	0.86
12	24	10.50	12.50	0.00

Table 9: Reference environmental conditions used in this thesis.

With the reference design and environmental conditions analyzed, the optimization problem can be presented. Before diving deeper into the optimization, the analysis tools that have been used throughout this thesis are demonstrated. This demonstration is needed to give an overview of the tools that are implemented inside the optimization workflow. Firstly, RAFT and ROSCO are presented, followed by the surrogate models.

4.3 RAFT

RAFT (Response Amplitudes of Floating Turbines) is a floating wind turbine frequency-domain analysis model. This section reports the key concepts utilized by RAFT, based on Hall's publication on RAFT (Hall et al., 2013 [49]). As a frequency-domain model, RAFT computes the steady-state

response of the substructure based on the frequency of the excitation loads. To compute the steady-state solution of a problem, RAFT takes the following actions:

- RAFT conducts an unloaded analysis on the substructure, analyzing its unloaded equilibrium position. Based on this analysis, the static properties of the substructure (substructure mass, mooring lines' equilibrium position and tension, hydrostatics etc.) are computed.

- The model's modes and natural frequencies are calculated.

- The external loads are divided into the mean (quasi-static) and frequency-dependent (dynamic) loads.

- For each load case defined in RAFT, the quasi-static and dynamic equilibrium is solved. Firstly, RAFT computes the mean responses of the substructure (based on the mean loads) and derives the system matrices about that mean responses. Finally, the linearized dynamic problem (based on the dynamic loads) is solved in the frequency domain.

The floating structure is simulated as a 6 degrees of freedom (DOFs) model. The model's DOFs are surge, sway, heave, roll, pitch and yaw. Additionally, the rotor controller is included in the model, including an additional DOF, the rotational speed of the rotor.

More specifically, the model calculates the response of the floater, $\hat{\boldsymbol{\xi}}(\omega)$, based on Equation 98:

$$\left(-\omega^2 \left[\mathbf{M}_{struc} + \mathbf{A}_{sub}(\omega) + \mathbf{A}_{\text{aero}}(\omega)\right] + i\omega \left[\mathbf{B}_{sub}(\omega) + \mathbf{B}_{\text{aero}}(\omega)\right] + \mathbf{C}_{struc} + \mathbf{C}_{\text{moor}}\right) \hat{\boldsymbol{\xi}}(\omega) = \hat{\mathbf{f}}(\omega) \tag{98}$$

where M, A, B and C are the mass, added mass, damping and restoring system matrix, respectively. $\hat{\mathbf{f}}(\omega)$ is the excitation force applied in each degree of freedom. The following paragraphs summarise the procedure adopted to assemble the elements of the equation, an overview of which is illustrated in Figure 24.
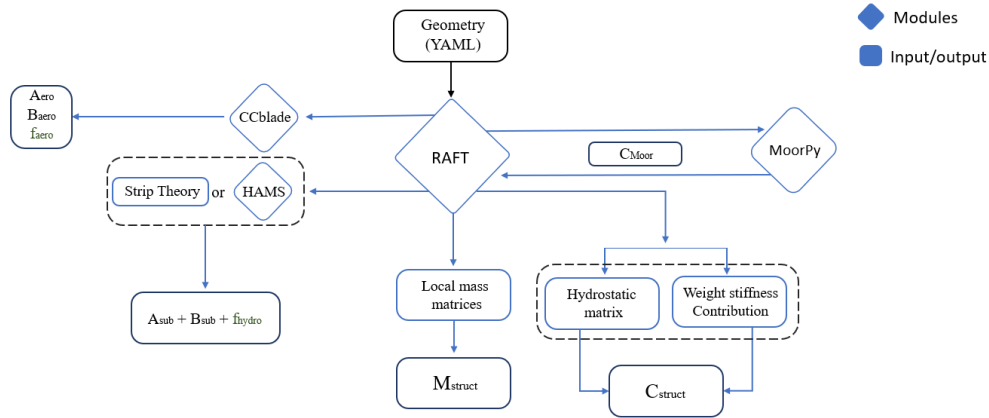
Figure 24: RAFT overview diagram

4.3.1   Mass matrix

The local six-by-six mass and inertia matrix are computed based on the geometrical and structural (material distribution) properties specified in the input file. The local matrices are then transformed with respect to the global reference point of the platform-wind turbine system and summed up, leading to the mass matrix of the structure M_{struct}.

Following the mass matrix, the restoring properties of the floater are calculated.

4.3.2 Stiffness matrix

The total stiffness of the system is the result of the floater's stiffness C_{struct} and the Mooring lines' stiffness C_{moor}. Regarding the structure's stiffness, it consists of the hydrostatic stiffness, K_{hydro}, and the contributions of the structure's weight distribution. K_{hydro} is calculated for every member with respect to the global reference point as:

$$\mathbf{K}_{hydro,i} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \rho g A_{wp} & \rho g A_{wp} y_{wp} & -\rho g A_{wp} x_{wp} & 0 \\ 0 & 0 & \rho g A_{wp} y_{wp} & \rho g I_{xx} + \rho g V z_{cb} & -\rho g A_{wp} x_{wp} y_{wp} & 0 \\ 0 & 0 & -\rho g A_{wp} x_{wp} & -\rho g A_{wp} x_{wp} y_{wp} & \rho g I_{yy} + \rho g V z_{cb} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{99}$$

where A_{wp} is the water plane area, x_{wp} and y_{wp} are the coordinates of the water plane, V is the displaced volume of the member, and z_{cb} is the vertical location of the volume's center. Additional, I_{xx} and I_{yy} are the water plane area moments of inertia about the x and y directions, respectively.

In Equation 99, if a member is fully submerged, the water plane area A_{wp} is set to zero. Furthermore, members that don't interact directly with the water volume (tower, RNA assembly), are not included in the calculations. When the stiffness of every member is calculated, the individual stiffness is summed to a total hydrostatic stiffness matrix. What is left is to include the structure's weight contribution in pitch and roll, which is equal to $-m_{\text{total}} Z_{cg}$. The result of those steps is the platform's stiffness matrix C_{struct}.

Simulating the mooring lines in RAFT is achieved by calling a quasi-static mooring analysis sub-component, Moorpy [55]. The mooring lines, floater's hydrostatic properties and load case are passed from RAFT to MoorPy. MoorPy then calculates the mean offset, $\bar{\xi}$, the mooring equilibrium force, $\bar{f}_{moor}(\bar{xi})$, and the mooring lines stiffness matrix, C_{moor}. Moreover, MoorPy computes the tension of each mooring line and the Jacobian of the mooring line tension, with respect to the platform's DOF.

4.3.3 Hydrodynamic forces

There are two ways to calculate the hydrodynamic properties in RAFT: by strip theory or potential flow theory. Potential flow theory is solved by coupling RAFT with the HAMS module. HAMS (Liu, 2019 [56]) is a model calculating hydrodynamic forces based on potential flow theory. In order for that, RAFT has a built-in tool that discretizes the floater geometry, creating a mesh. This process takes place automatically before HAMS is called. HAMS computes the added mass and damping matrices for the system and the waves' excitation forces.

In the case of strip theory, the hydrodynamic force applied on a strip (Δf) is based on Morison's equation:

$$\Delta f = [\underbrace{\rho \left(1 + C_a\right) \frac{\pi}{4} D^2 \dot{u}}_{\text{inertia}} - \underbrace{\rho C_a \frac{\pi}{4} D^2 \dot{v}}_{\text{added mass}} + \underbrace{\frac{1}{2} \rho C_d D (u - v)|u - v|}_{\text{drag}}] \Delta l \tag{100}$$

where D is the member's characteristic width, Δl is the length of the strip, C_a and C_d are the user-defined added mass and drag coefficients, respectively, and u and v are the wave and body velocity. The wave-induced velocities and accelerations are calculated based on Airy wave theory for the case of intermediate water depth.

4.3.4 Added mass and damping matrices

As already mentioned, if potential flow theory is applied, HAMS computes the added mass and hydrodynamic damping matrices based on the potential flow solution of the discretized substructure. If strip theory is applied, the local added mass for each strip is obtained by Equation 101:

$$\mathbf{A}_L = \rho \Delta V \left(C_{a,p1} \mathbf{p}_1 \mathbf{p}_1^T + C_{a,p2} \mathbf{p}_2 \mathbf{p}_2^T \right) \tag{101}$$

in this equation, p_1 and p_2 are the orthogonal transverse unit vectors, and $C_{a,p1}$ $C_{a,p2}$ are the corresponding (user-defined) added mass coefficients. When the calculations are done, the local matrices are transformed to the global (platform) reference coordinate system and summed, leading to the total added mass of the floater, A_{sub}.

Lastly, the local damping matrix for each strip is computed as:

$$\mathbf{B}_L = \frac{1}{2} \rho \sqrt{\frac{8}{\pi}} \left(\sigma_{u,q} A_q C_{d,q} \mathbf{q} \mathbf{q}^T + \sigma_{u,p1} A_{p1} C_{d,p1} \mathbf{p}_1 \mathbf{p}_1^T + \sigma_{u,p2} A_{p2} C_{d,p2} \mathbf{p}_2 \mathbf{p}_2^T \right) \tag{102}$$

where A_q is the effective axial drag area of the strip, $C_{d,q}$ is the specified axial drag coefficient, q is the axial unit vector and σ_u is the standard deviation of the relative water velocity. Similarly to mass and added mass, the local damping matrices are transform to the global coordinate system and summed, giving B_{sub} of Equation 98.

4.3.5 Aerodynamics and control

Aerodynamic loads are simulated on CCblade, a steady-state BEM solver known for its efficiency and robustness at always converging to a solution (Ning et al., 2013 [57]). The blade geometry, consisting of constant-section elements, is imported by RAFT. RAFT also provides CCBlade with the environmental wind speed, wind shear and the wind turbine's rotor speeds and pitch angles as functions of the wind speed. Based on the inputs, CCBlade calculates the steady-state regime's thrust, torque and power output. These quantities are used to compute the annual energy production and the steady equilibrium position, while their derivatives are utilized to compute the aerodynamic contribution to the system. It is important to note, though, that even if in this thesis CCBlade is coupled with RAFT for solving the aerodynamics of the rotor, RAFT can also conduct a hydrodynamic and mooring analysis of a floating wind turbine substructure without including any aerodynamic elements to the problem. Respectively, RAFT can also solve an aero-hydrodynamic problem without considering any controller.

In the case of disabling the rotor controller, the rotor is assumed to spin with a constant pitch angle and velocity. Based on the thrust derivative ($\frac{\partial T}{\partial U}$), the fore-aft aerodynamic damping coefficient is calculated and by multiplying it with the rotor-averaged wind speed spectrum at hub-height, the aerodynamic force amplitude is obtained (as a function of frequency). The final step is to transform the damping and excitation force matrix with respect to the global coordinate system (centre of floater), giving the total damping matrix $B_{aero}(\omega)$ and aerodynamic excitation force $f_{aero}(\omega)$

When the controller is functional, RAFT considers the effect of fluctuating rotor speed, $\Omega(\omega)$, and blade pitch, $\beta(\omega)$, on the aerodynamic added mass, damping and force. In this case, thrust force, T, is calculated as:

$$T = \bar{T} + \frac{\partial T}{\partial u} \Delta(U - \dot{x}) + \frac{\partial T}{\partial \omega} \Delta\Omega + \frac{\partial T}{\partial \beta} \Delta\beta \tag{103}$$

where U is the rotor-average wind speed, and $\dot{x}$ is the fore-aft nacelle velocity. Additionally, Generator torque, τ_g, is given by:

$$I_r \dot{\Omega} = Q_U \Delta (U - \dot{x}) + Q_\Omega \Delta \Omega + Q_\beta \Delta \beta - N_g \Delta \tau_g \qquad (104)$$

where I_r is the rotor's mass moment of inertia, and N_g is the gearbox ratio.

Regarding the pitch and torque control inputs of Equation 103 and Equation 104, Δ_β and Δ_{τ_g}, they are given by:

$$\Delta \beta = k_{P\beta} \Delta \Omega + k_{I\beta} \int \Delta \Omega dt + k_{Px} \dot{x} \qquad (105)$$

and

$$\Delta \tau_g = k_{p,\tau} \Delta \Omega + k_{I,\tau} \int \Delta \Omega dt \qquad (106)$$

These inputs are calculated based on a proportional (k_p) and an integral (K_I) term. For the pitch controller, an additional nacelle-velocity feedback gain term is introduced (k_{p_x}), to diminish negative feedback. These equations follow the same philosophy as Equation 49, as they are both utilizing a PI controlling strategy.

In the frequency domain, the above-mentioned equations are combined to give the variation of the frequency-dependent thrust force:

$$T(\omega) = T_U U(\omega) - (T_U - k_{Px} T_\beta) i\omega x(\omega) - H_{QT}(\omega) \left[Q_U U(\omega) - (Q_U - k_{Px} Q_\beta) i\omega x(\omega) \right] \qquad (107)$$

where $H_{QT}(\omega)$ is the transfer function from the frequency-dependent rotor torque to the thrust, given by:

$$H_{QT}(\omega) = \frac{(T_\Omega + k_{P\beta} T_\beta) i\omega + k_{I\beta} T_\beta}{\omega^2 I_r + i\omega (Q_\Omega + k_{P\beta} Q_\beta - N_g k_{P\tau}) + k_{I\beta} Q_\beta - N_g k_{I\tau}} \qquad (108)$$

With $T(\omega)$ and $H_{QT}(\omega)$ defined, the aerodynamic added mass, damping and turbulence wind excitation matrices (with respect to the hub-height) are given by:

$$a_{\text{aero}}(\omega) = \Re \left\{ \frac{1}{i\omega} \left[T_U - k_{Px} T_\beta - H_{QT}(\omega) (Q_U - k_{Px} Q_\beta) \right] \right\} \qquad (109)$$

$$b_{\text{aero}}(\omega) = \Re \left[T_U - k_{Px} T_\beta - H_{QT}(\omega) (Q_U - k_{Px} Q_\beta) \right] \qquad (110)$$

$$\hat{f}_{\text{aero}}(\omega) = (T_U - H_{QT}(\omega) Q_U) U(\omega) = H_{Uf}(\omega) U(\omega) \qquad (111)$$

These terms are then transformed into the platform reference point by including their effect on surge and pitch. From that transformation, the aerodynamic mass ($A_{aero}(\omega)$), damping ($B_{aero}(\omega)$) and excitation ($\hat{f}_{aero}$) matrices are defined.

4.4 ROSCO

When exploring alternative substructure designs, adopting a one-controller-fits-them-all approach can not guarantee that each design configuration's controller operates in an optimum way. In order to ensure optimum controller operation, customizing (tuning) the controller settings according to the substructure characteristics is crucial. Especially in the case where the controller's reaction

time is similar to the floating wind turbine's natural pitch frequency, the results can deviate significantly from the case of a tuned controller. More importantly, power quality, which is extensively investigated in this research, is greatly affected by the controller settings, due to its strong connection with the pitch and torque control.

In order to overcome this obstacle, each new designs' controller is tuned by coupling RAFT with NREL's ROSCO (Reference OpenSource COntroller, Abbas et al., 2021 [58]), an open-source framework providing a generic tuning process. As a reference controller framework, ROSCO is designed for VSVR controllers, with both the torque and pitch controller's response computed by utilizing PI gains (see Section 2.4.3). Moreover, when analyzing floating wind turbines, it incorporates a nacelle feedback gain term (also referred to as floating gain in the ROSCO algorithm) to tackle negative feedback in Region III. It also features options for additional controlling techniques, like peak-shaving (reducing thrust force close to the rated wind speed area).

Coupling RAFT and ROSCO tunes both the blade pitch and torque generator gains. For the case of a floating wind turbine, the floating gain term is also tuned. Especially for the generator torque and floating gain, implementing ROSCO has significant effects on the operation of the wind turbine. In the original RAFT workflow, the proportional and integral torque gains and the floating gain terms are constant. By integrating ROSCO, the torque and floating gains can be expressed as functions of the blade pitch (or wind speed), increasing the simulation's fidelity (the controller behaves more realistically).

4.4.1 ROSCO workflow

ROSCO's workflow is shown in Figure 25. The process starts with a YAML and a performance file in the ROSCO toolbox. The YAML file contains the characteristics of the turbine and specific controller tuning parameters. The performance files are OpenFAST simulation results that are related to the performance of the wind turbine, such as the corresponding pitch angle θ, C_p, C_t, C_q for specific wind speeds. ROSCO processes the data input and prints the tuned controller properties. For this thesis, the properties processed by ROSCO are the proportional and integral gains of torque and pitch control and the floating gain term.
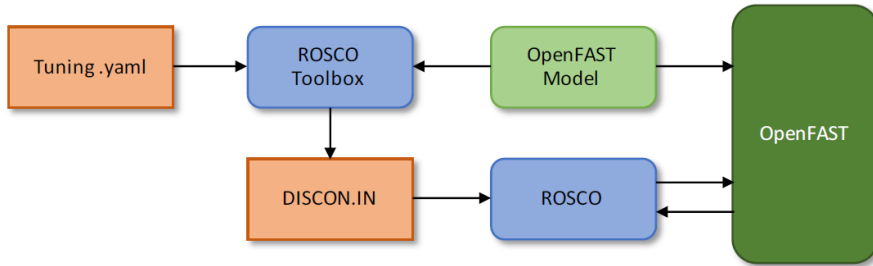
Figure 25: ROSCO workflow diagram (Abbas et al., 2022 [14])

More in detail, the reference tracking PI controller of ROSCO is defined as:

$$y = k_p d\omega_g + k_i \int_0^T d\omega_g dt \qquad (112)$$

where $d\omega_g$ is the shaft speed tracking error. In torque control applications, $y = d\tau_g$, and for pitch control applications, $y = d\beta$, which leads to Equation 49. Regarding the additional floating gain term, when implemented into the system, the reaction of blade pitch control becomes:

$$d\beta = k_p d\omega_g + k_i \int d\omega_g dt + k_{\beta_{\text{float}}} \, d\dot{\phi} \tag{113}$$

where $k_{\beta,float}$ is a proportional gain term, and $\dot{\phi}$ is the tower top velocity in the fore-aft direction.

In order to tune the aforementioned gains, ROSCO firstly explores the potential C_p factors that can be achieved for specific TSR and pitch angle combinations. As a default choice, ROSCO utilizes the steady-state blade element momentum solver CCblade to calculate the C_p surface map. Based on that exploration, a C_p surface map is created, such as the one illustrated in Figure 26. This surface map is the one primarily used to tune the controller parameters. Essentially, the tuning process optimizes the parameters so that, for below-rated conditions, $C_{p,max}$ is preserved. For above-rated conditions, where TSR reduces (only wind speed increases, while the blades angular velocity stays constant), the controlling parameters should be able to achieve the maximum feasible C_p, which is indicated in the red dashed line in Figure 26. More analytical details on the specific techniques used for tuning the controller are given in the Abbas et al. publication, devoted to the ROSCO framework (Abbas et al., 2021 [14]). Additionally, an overview of the way that RAFT and ROSCO are coupled in this thesis is given in Appendix D
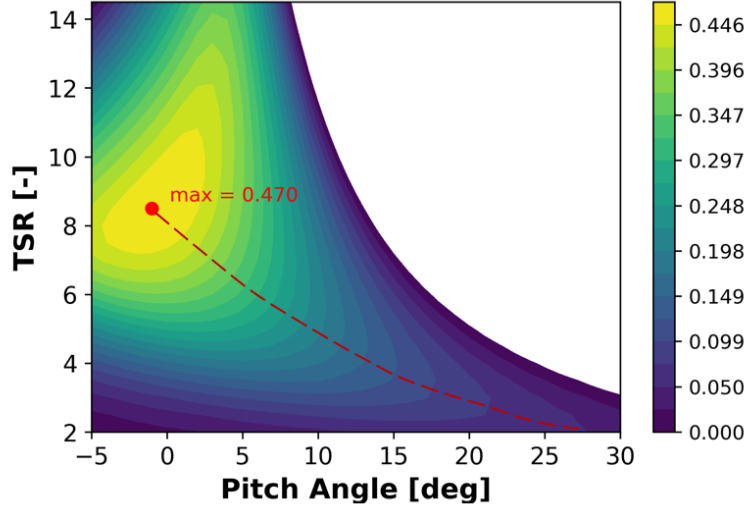
Figure 26: IEA 15 MW reference wind turbine C_p surface, created in ROSCO.

4.5   Design variables

For this project, four design parameters are investigated: The outer column diameter D_o, The radial distance between the inner and the outer columns L_o, The mooring lines' total length S_t, and the mooring chains diameter d_c. The reason for choice is based on the fact that those parameters are expected to significantly affect both the hydrodynamic behaviour and the cost of the floater. Further to that, by reviewing the existing literature, it becomes evident that, in most cases, research is either focused on the floater or the mooring parameters. Additionally, there are three design parameters that are derived from the design variables. Those are the diameters of the pontoons (D_{pon}), the radial distance of the anchor (X_{tot}), and the wall thickness of the pontoons (t_{pon}). The above-mentioned design parameters, together with the ones kept constant, are summarized in Figure 27.
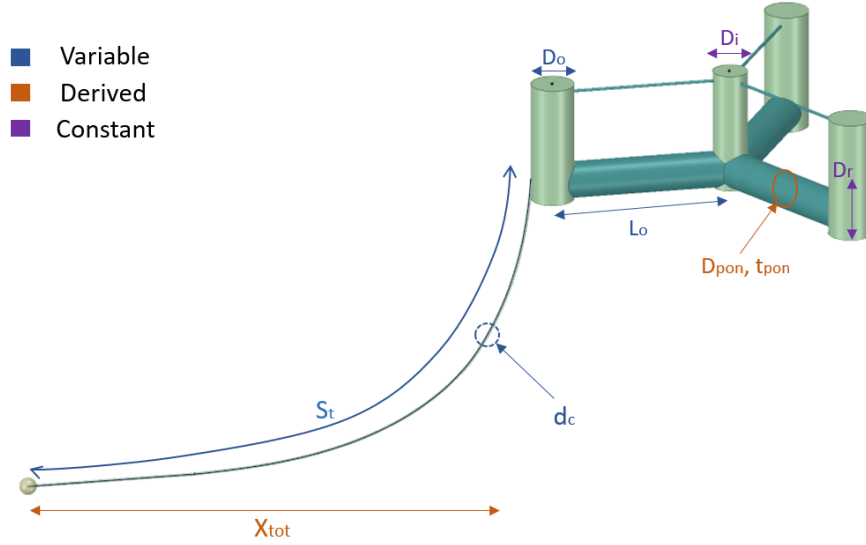
Figure 27: Design parameters analysis sketch.

To define the boundaries of the design space, three geometrical constraints are taken into account. Unlike the optimization constraints (Section 4.8), these geometrical constraints make every design violating them impractical, regardless of the environment in which the substructure is located.

Mathematically, the three constraints are defined as:

$$L_o - (\frac{D_o}{2} + \frac{D_i}{2}) > 0 \tag{114}$$

$$D_{pon} < T_d \tag{115}$$

$$M_{bal} > 0 \tag{116}$$

where L_i is the pontoon length and D_{pon} is the diameter of the pontoon. Starting from Equation 115, The first constraint ensures that the outer and inner columns don't touch each other due to the variability of their distance and diameter, which would lead to a result similar to Figure 17. This constraint is violated for design configurations of large D_o values combined with a short pontoon length, meaning a small L_o value. In Equation 115, it is ensured that the pontoons are fully submerged. As the pontoons' diameter depends on the varying outer column diameter, D_o, this constraint affects designs with large outer diameters. Finally, the constraint defined by Equation 116 regulates the mass of the solid ballast filling the outer columns, which should always be positive. This constraint is violated for designs of small outer column diameters, as that leads to negative vertical buoyancy.

Based on those three constraints, the feasible space is defined. To do so, an abundance of geometries is produced by the geometry generator for a very wide L_o and D_o range. An overview of the geometry generator algorithm is given in Appendix E. Both the L_o and D_o values range are divided into equal intervals, creating a grid. Every time a design configuration violates one of the constraints, it is identified as an unfeasible design. The resulting contour plot from this process is shown in Figure 28. As expected, large D_o values combined with small values of L_o lead to the collision of the outer and inner columns. Also, for very small values of D_o, the structure's steel weight over exceeds the water displaced volume, resulting in the negative ballast constraint. Finally, for outer columns past a certain diameter, the pontoons become large and, even in an

equilibrium position, they emerge out of the surface level.

According to the feasible design space of Figure 28, it becomes clear that the available design freedom is more than what is needed for the optimization problem. Even if there was no computational cost during the optimization process, the desired optimized designs should still have similar size as the reference design. Considering that, it is reasonable to optimize the floater's design parameters within the [9.8, 15.8] range for D_o and [47, 57] for L_o.
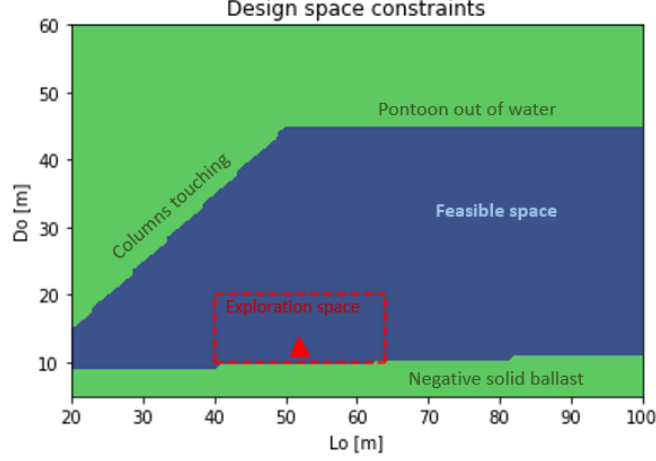
Figure 28: Design space constraints and under-exploration space (red box). Red triangle indicates the reference design.

4.6 Derived design parameters

Apart from the design parameters, there are specific design properties of the substructure that are derived from the design parameters and environmental conditions. These are the anchor radial distance X_{tot}, the pontoon wall thickness t_{pon}, and the cylindrical pontoon diameter D_{pon}.

Starting from the cylindrical pontoon diameter, it is already mentioned in Section 4.1 that its value is related to the outer column diameter, D_o. That way, for larger outer columns, a part of the additional buoyancy would be counteracted by the additional water ballast volume of the bigger pontoons, reducing the solid ballasts needed. In order to apply the same logic to the cylindrical pontoons, the two members should have the same volume. Since the length of both pontoon geometries is the same, only their cross-sectional areas need to be equal:

$$A_{rect} = A_{cyl} \implies D_{pon} = 2\sqrt{\frac{7D_o}{\pi}} \tag{117}$$

Furthermore, the cylindrical pontoon's central axis coordinates are modified so that its bottom is at the same elevation as the bottom of the inner and outer columns.

Regarding the anchor radius X_{tot}, its value is derived by the location depth. Specifically, in the reference design, X_{tot} is 790.6 m for a reference depth of 200 m. By accounting for the fairlead point located 14 m below the sea level, in case a different water depth is used, the ratio between the fairlead-seabed distance and X_{tot} is kept constant, given as:

$$\frac{Depth - h_{fair}}{X_{tot}} = \frac{Depth_{ref} - h_{fair,ref}}{X_{tot}} = 0.239 \tag{118}$$

4.6.1 Pontoons wall thickness

The pontoons wall thickness is estimated based on a preliminary static analysis and application of Eurocode 3 for shell elements [59]. This analysis accounts for the static loads applied on a pontoon, calculates the maximum resulting stresses, and ensures that the wall of the pontoon, modelled as a shell, is thick enough to withstand the calculated stresses. Even though it does not account for the dynamic loads, it still results in a more realistic wall thickness, adding cost with the variation of the design variables instead of a constant wall thickness.

To calculate the stresses applied on each pontoon, the problem shown in Figure 29 is analyzed. In this analysis, the following assumptions are made:

- The pontoon is simulated as a cantilever cylindrical beam, fixed at the connection with the inner column and free at the connection with the outer column.

- The top brace does not contribute to the loads analysis.

- The hydrostatic pressure applied on the pontoon is equal to the hydrostatic pressure at the bottom of the pontoon. Even though this is a conservative assumption, the stresses resulting from this load do not significantly affect the wall thickness.

- The water ballast inside the pontoon does not apply any internal pressure to the walls.

As illustrated in Figure 29, there are three loads considered on the floater: A vertical and a horizontal concentrated load applied on the outer columns, F_v and F_H, and a distributed load applied throughout the pontoon length. The concentrated loads are the result of the buoyancy and mooring forces applied on the column, given by Equation 119 and Equation 120:

$$F_v = max\{V_{moor,max}, F_{buoy}\} \tag{119}$$

and

$$F_h = H_{moor,max} \tag{120}$$

where $V_{moor,max}$ and $H_{moor,max}$ are the maximum vertical and horizontal loads that the mooring line can apply (as calculated by RAFT). These loads are equal to the static analysis (mean) load and three times the standard deviation of the dynamic analysis load. Mathematically:

$$F_{max} = F_{mean} + 3\sigma_{F,dyn} \tag{121}$$

Finally, F_{buoy} is the net buoyancy force applied on the end of the pontoon, calculated as:

$$F_{buoy} = T_d \cdot \rho_w \cdot \pi \cdot D_{out}^2 \cdot g/4 - M_{bal}g - M_{steel,col} \cdot g \tag{122}$$

where T_d is the draft of the floater (20 m), ρ_w is the sea water density (1024 kg/m^3), D_{out} is the outer column diameter, and M_{bal} is the mass of the solid ballasts in one outer column.

Regarding the distributed load, based on the hydrostatic pressure assumption, it is given as:

$$q = D_{pon} \cdot \rho_{water} \cdot g \cdot T_d \tag{123}$$

With the applied loads known, the maximum stresses resulting from the static loads can be computed. These stresses depend on the maximum meridional (axial), shear and circumferential (hoop) stresses. For a cantilever beam loaded as shown in Figure 29, the maximum stresses are concentrated on the root of the pontoon (connected to the inner column). The maximum meridional stress, $\sigma_{x,Ed}$, is equal to:
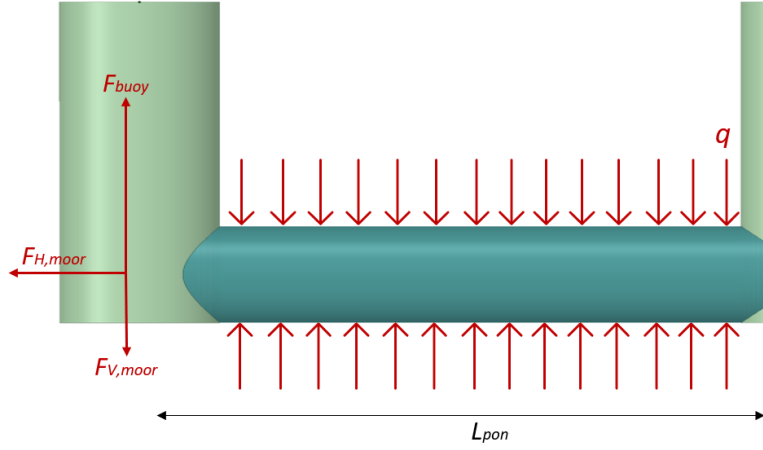
Figure 29: Loads analysis overview

$$\sigma_{\chi,Ed} = \frac{F_h}{A_s} + \frac{M_y}{I}\frac{D_{pon}}{2} \tag{124}$$

where M_y is equal to the moment applied on the root of the pontoon, A_s is the cross-section area of the pontoon, and I is the pontoon's moment of inertia, given by:

$$I = \pi \frac{D_{pon}^4}{4} - \pi \frac{Di_{pon}^4}{4} \tag{125}$$

where Di_{pon} is the inner pontoon diameter. The maximum shear stress in a hollow cylinder is given by:

$$\tau_{\chi\theta,Ed} = \frac{F_V \cdot Q_{max}}{I \cdot t_{pon}} \tag{126}$$

where Q is the first moment of area, which, for a case of a circular section is maximized according to Equation 127:

$$Q_{max} = \frac{3 \cdot \pi \cdot D_{pon} \cdot (D_{pon}^2 - Di_{pon}^2)}{160} \tag{127}$$

To calculate the hoop stress caused by the distributed load q, a section of the pontoon is considered, as shown in Figure 30. According to the forces indicated, the hoop stress is equal to:

$$\sigma_{teta,Ed} = \frac{Pe \cdot D_o}{2 \cdot t_{pon}} \tag{128}$$

As already indicated, I, A_s, D_i, σ_θ and τ_{max} depend on the pontoon wall thickness, t_{pon}. The minimum required wall thickness can be found by considering that the combination of the applied stresses should be less than the combination of the resistance stresses the pontoon is designed to hold. Since the pontoon is a shell element, the stresses combination criteria becomes:

$$\left(\frac{\sigma_{x,Ed}}{\sigma_{x,Rd}}\right)^{k_x} - k_i \left(\frac{\sigma_{x,Ed}}{\sigma_{x,Rd}}\right)\left(\frac{\sigma_{\theta,Ed}}{\sigma_{\theta,Rd}}\right) + \left(\frac{\sigma_{\theta,Ed}}{\sigma_{\theta,Rd}}\right)^{k_\theta} + \left(\frac{\tau_{x\theta,Ed}}{\tau_{x\theta,Rd}}\right)^{k_z} \leq 1 \tag{129}$$
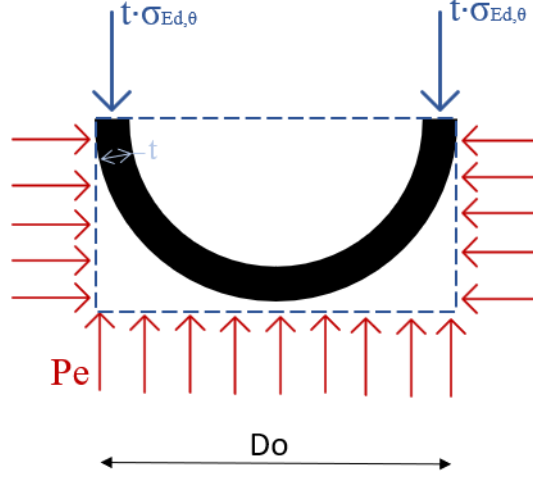
Figure 30: Hoop stresses analysis

In Equation 129, Ed indicates the applied stresses, and Rd indicates the resistance stresses (the maximum stresses the component can hold). According to EN 3, $\sigma_{\chi,Rd}$, $\tau_{\chi\theta,Rd}$ and $\sigma_{\theta,Rd}$ are given by:

$$\sigma_{\mathrm{x,Rd}} = \sigma_{\mathrm{x,Rk}}/\gamma_{\mathrm{M1}} \tag{130}$$

$$\tau_{\mathrm{x}\theta,\mathrm{Rd}} = \tau_{\mathrm{x}\theta,\mathrm{Rk}}/\gamma_{\mathrm{M1}} \tag{131}$$

$$\sigma_{\theta,\mathrm{Rd}} = \sigma_{\theta,\mathrm{Rk}}/\gamma_{\mathrm{M1}} \tag{132}$$

with $\gamma_{M1} = 1.1$ and:

$$\sigma_{x,Rk} = \chi_x f_{\mathrm{yk}} \tag{133}$$

$$\tau_{\mathrm{x}\theta,\mathrm{Rk}} = \chi_\tau f_{\mathrm{yk}}/\sqrt{3} \tag{134}$$

$$\sigma_{\theta,\mathrm{Rk}} = \chi_\theta f_{\mathrm{yk}} \tag{135}$$

In Equation 133 to Equation 135, f_{yk} is the yield stress of steel, which can be (conservatively) assumed as $250\ MPa$, and χ_χ, χ_τ and χ_θ are the buckling reduction factors. These three reduction factors are functions of the pontoon's slenderness $\bar{\lambda}$ and are given by:

$$\chi = 1 \text{ when } \bar{\lambda} \leq \bar{\lambda}_0 \tag{136}$$

$$\chi = 1 - \beta \left(\frac{\bar{\lambda} - \bar{\lambda}_0}{\bar{\lambda}_\mathrm{p} - \bar{\lambda}_0} \right)^\eta \text{ When } \bar{\lambda}_0 < \bar{\lambda} < \bar{\lambda}_\mathrm{p} \tag{137}$$

$$\chi = \frac{\alpha}{\bar{\lambda}^2} \text{ when } \bar{\lambda}_\mathrm{p} \leq \bar{\lambda} \tag{138}$$

where α is the elastic imperfection reduction factor, β is the plastic range factor, n is the interaction exponent and $\bar{\lambda}_0$ is the squash limit relative slenderness. With the exception of β, which is equal to 0.6, the rest of these parameters are different for Equation 133 to Equation 135. Together with the slenderness λ, they depend on the dimensions of the pontoon and, according to each design's parameters, they are defined by different formulas, which can be found in Appendix F.

With the resistance (Rd) stresses computed, the 4 unknown terms of Equation 129 left are the

buckling interaction parameters, k_χ, k_z, k_i and k_θ, which are related to the buckling reduction factors as:

$$k_\text{x} = 1,25 + 0,75\chi_\text{x} \tag{139}$$

$$k_\tau = 1,75 + 0,25\chi_\tau \tag{140}$$

$$k_\theta = 1,0 + \chi_\theta^2 \tag{141}$$

$$k_\text{i} = (\chi_x\chi_\theta)^2 \tag{142}$$

With every parameter of Equation 129 computed, the equation is solved in order to compute the minimum pontoon wall thickness that fulfills the check. Also, even though the safety factor is set as 1, the analysis is quite conservative since a low yield-stress normal quality steel is selected, with affects the stress resistance of the component.

4.7   Optimization objectives

This thesis aims to investigate the trade-offs between substructure cost and power quality of a floating wind turbine. Thus, the optimization workflow's objective functions are the cost of the substructure and the quality of the power output. By defining two objective functions, the optimization goal is not to obtain one single optimum design. On the contrary, the optimization goal is to produce a *Pareto front*, which is a set of non-dominated designs. A design is considered non-dominated if, for a given value of the one objective function, there is no other design that performs better in the other objective.

4.7.1   Substructure cost

The substructure's cost is evaluated according to:

$$Cost = C_{conc} \cdot M_{bal} + C_{steel} \cdot (M_{steel} + 3 \cdot W_{chain} \cdot S_t) + 3 \cdot C_{anch} \cdot M_{anch} \tag{143}$$

where $C_{conc} = 70\ eur/t$, $C_{steel} = 495\ eur/t$ (Trading Economics [60]), W_{chain} is the submerged weight per meter of the mooring line, S_t is the total mooring line length, M_{anch} is the mass of the anchor and $C_{anch} = 5 \cdot C_{steel}$. Even though the anchors are also made of steel, it is reasonable to assume that it is more expensive to manufacture an anchor compared to a floater and a mooring chain.

To calculate the mass of the anchor, it is assumed that the mooring lines are attached to a drag anchor. The anchor used is the Stevpris MK5 [61]. In order to calculate the anchor mass, the required Ultimate Holding Capacity is calculated, which is equal to:

$$UHC = 1.5 \cdot H_{moor,max} \tag{144}$$

where 1.5 corresponds to a safety factor and $H_{moor,max}$ is the horizontal load applied to the mooring line when fully suspended. Based on its UHC value, the mass of the anchor can be calculated based on the diagram of Figure 31. In this diagram, it is assumed that the soil is a medium clay (red line), based on which the UHC and weight of the anchor can be accurately expressed by:

$$UHC = 552.53 \cdot W^{0.92} \tag{145}$$

where W is the weight of the anchor in MT, and UHC is in KN.
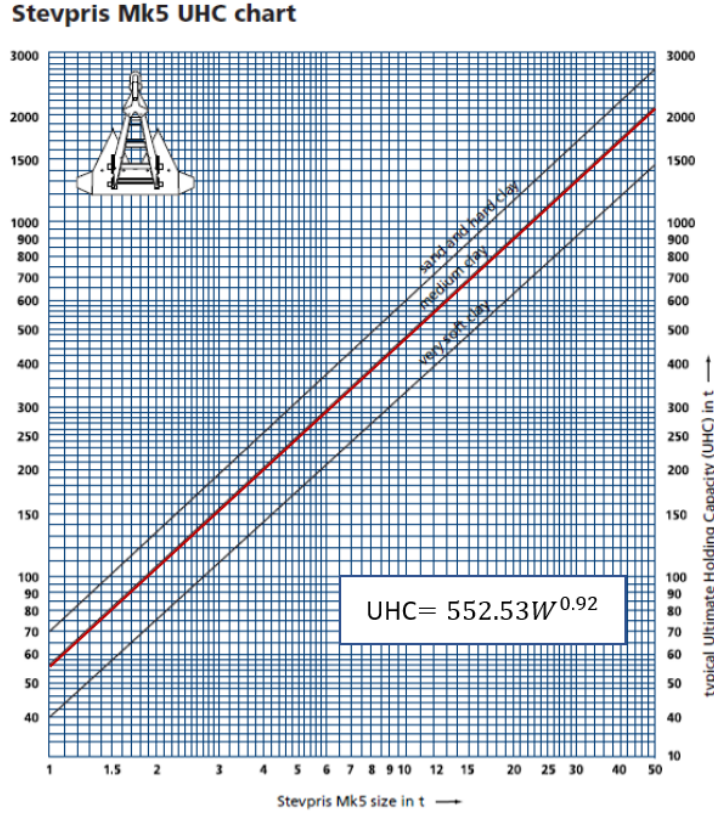
Figure 31: Stevpris MK5 UHC and mass relation diagram. Red line corresponds to the medium soil conditions reference.

4.7.2 Power quality

In this thesis, the quality of a wind turbine's power output is estimated by the produced power fluctuations. A proxy to estimate this objective is by calculating the standard deviation of the mechanical power produced by the turbine, σ_P, given by:

$$P = \tau_{gen} \cdot \omega_{rotor} \tag{146}$$

where τ_{gen} is the generator torque and ω_{rotor} is the angular velocity of the rotor. However, RAFT does not calculate the standard deviation of power directly. Instead, it provides as output the mean value μ and standard deviation σ of τ_{gen} and ω_{rotor}. Considering those outputs and Equation 146, power quality (power standard deviation) can be estimated as:

$$\sigma_P = \sqrt{\sigma_\omega^2 \sigma_\tau^2 + \mu_\omega^2 \sigma_\tau^2 + \mu_\tau^2 \sigma_\omega^2} \tag{147}$$

The limitation of this approximation is that Equation 147 requires that the two variables, ω and τ, are independent. In reality, this is an invalid assumption, as the rotor rotational velocity and the generator torque are closely related. Furthermore, for wind speeds close to the rated value, the behaviour of the power produced is expected to be non-linear. This is the reason why, in the reference environmental cases mentioned in Table 9, all the wind speeds are at least 1 m/s away from the rated wind speed (11.4 m/s).

Nevertheless, the power quality approach taken in this thesis is expected to produce reliable results. As discussed during the literature review (Section 3), in the few cases where power quality is investigated, only the standard deviation of the rotor angular velocity is taken into account. On the contrary, Equation 147 takes a more realistic approach, as especially for below-rated conditions,

the standard deviation of the generator torque is the significantly changing one.

Due to the definition of power quality, in the Results chapter, the results on power quality are analyzed based on the standard deviation of the power produced (σ_P). This comment is made to add consistency to the way results are analyzed. Specifically, a reduction in the standard deviation of power production indicates an improvement in power quality and vice versa. This clarification is important as many results are dependent on power quality.

4.8   Constraints

Defining constraints sets the limits needed to ensure that the optimizer does not converge to unfeasible solutions. In the following paragraph, 8 constraints are defined. Those constraints regulate the substructure's responses, unrealistic mooring designs, the natural frequencies of the substructure.

Regarding the responses' constraints, the maximum surge, maximum pitch and maximum nacelle acceleration are limited. The maximum response is computed as:

$$x_{max} = \bar{x} + 3 \cdot \sigma_x \tag{148}$$

where $\bar{x}$ is the mean value of the response x and σ_x is the standard deviation of the response for a given environmental case. For the case of surge, the maximum acceptable displacement is 30% of the water depth. For the nacelle acceleration, the maximum accepted value is $0.2g$, with g being the gravitational acceleration (9.81 m/s). Finally, the maximum accepted pitch response is $10 \deg$.

Regarding the mooring lines, the first constraint applied ensures that the mooring line's total length S_t is enough to connect the floater with the anchor. Mathematically, this constraint is defined as:

$$S_t > \sqrt{D^2 + X_{tot}{}^2} \tag{149}$$

Furthermore, The mooring chains should be able to withstand the maximum mooring tension applied on the mooring line. By utilizing Equation 77, the constraint becomes:

$$T_{moor,max} < w_{water}[103.84(44 - 0.08d_{chain})] \tag{150}$$

where $T_{moor,max}$ is the maximum mooring line tension, for a specific environmental case, given by:

$$T_{moor,max} = \bar{T}_{moor} + 3 \cdot \sigma_T \tag{151}$$

Apart from the mooring line's integrity, it is also important to ensure that there is no vertical force applied to the anchors. Since drag anchors are considered, the existence of vertical loads can lead to the displacement of the anchor, affecting the responses of the turbine. In order to ensure the anchor's stability, for a fully suspended line and for a given mooring tension, the minimum (fully suspended) chain length S_t needed to counteract the tension's vertical component is derived by Equation 78, which gives:

$$S_t > D \cdot \sqrt{2\frac{T}{w_{water}D} - 1} \tag{152}$$

Finally, it is important that the natural frequency of the substructure (for every DOF) is not in the same range as the RNA's 1P and 3P frequency ranges determined by the rotor rotational speed. The 1P and 3P frequency ranges of the IEA 15 MW are demonstrated in Figure 32. According to that, and by a 5% safety margin away from the frequency ranges, the natural frequency constraints are given by:

$$Fr_{i,nat} \leq 0.95 \cdot 1P_{min} \text{ and } Fr_{i,nat} \geq 1.05 \cdot 1P_{max} \tag{153}$$

and

$$Fr_{i,nat} \leq 0.95 \cdot 3P_{min} \text{ and } Fr_{i,nat} \geq 1.05 \cdot 3P_{max} \tag{154}$$

where i corresponds to the 6 DOFs of the substructure.
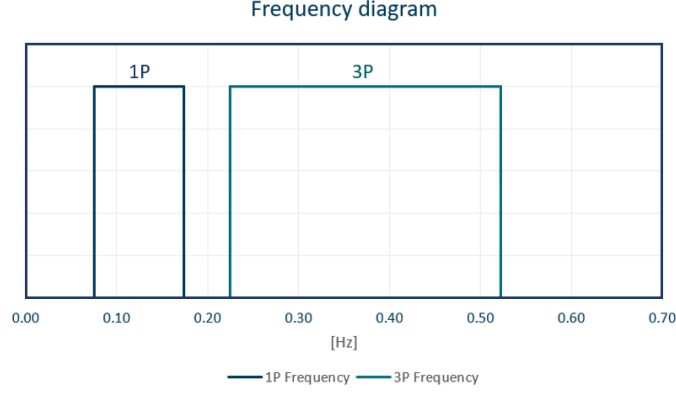
Figure 32: Frequency range of the 1P and 3P blade passing frequencies.

With the design parameters, constraints and objective functions defined, the optimization problem is formulated and given in Equation 155:

$$
\begin{array}{ll}
\text{minimize} & \sigma_P(D_o, L_o, S_t, d_c) \\
& Cost(D_o, L_o, S_t, d_c) \\[2mm]
\text{by varying} & 9.8 \leq D_o \leq 16 \\
& 48 \leq L_o \leq 57 \\
& 780 \leq S_t \leq 900 \\
& 0.080 \leq d_c \leq 0.180 \\[2mm]
\text{subject to} & 0.3D - \chi_{1,max} \leq 0 \\
& \chi_{5,max} - 10 \leq 0 \\
& \ddot{\chi}_{5,max} - 0.2g \leq 0 \\
& \sqrt{D^2 + X_{tot}{}^2} - S_t \leq 0 \\
& T_{moor,max} - w_{water}[103.84(44 - 0.08d_{chain})] \leq 0 \\
& D \cdot \sqrt{2\frac{T}{w_{water}D} - 1} - S_t \leq 0 \\
& Fr_{i,nat} \leq 0.95 \cdot 1P_{min} \text{ , } 0.105 \cdot 1P_{max} \leq Fr_{i,nat} \\
& Fr_{i,nat} \leq 0.95 \cdot 3P_{min} \text{ , } 0.105 \cdot 3P_{max} \leq Fr_{i,nat}
\end{array}
\tag{155}
$$

4.9 Optimization algorithm: The NSGA-II algorithm

Equation 155 contains all the characteristics of the optimization problem. That way, every individual design has a unique identity, given by its $[D_o, L_o, S_t, d_c]$ parameters, is either feasible or not feasible, and is evaluated based on its cost and the power quality it offers. Since it is impossible to investigate every possible design combination, what is missing from the optimization is a well-structured strategy that, based on the limited number of designs evaluated, indicates which

designs should be evaluated next. This strategy is implemented into the system in the form of the optimization algorithm.

In this optimization, two objectives are investigated. For that reason, the optimization's algorithm chosen is the Non-Dominating Sorting Genetic (NSGA-II) [62] algorithm, as it is a gradient-free, well-tested algorithm in the scientific community.

Essentially, the NSGA-II algorithm is a genetic algorithm with a specific selection process. The basic steps followed by the algorithm are shown in Figure 33. According to it, the NSGA-II optimization process can be divided into the following stages:

- For the generation $t = 0$, an initial population P_t of N individuals is randomly selected. Based on P_t, an offspring population Q_t is produced.

- The two populations P_t and Q_t are combined, creating a population of size 2N.

- The individuals of the population are then classified in several fronts based on a non-dominated sorting algorithm. In this sorting method, an individual dominates another one only if it performs better in every objective function. In this case, a design is classified higher than another one only if it is both cheaper and has a lower power standard deviation. Simultaneously, the crowding distance of the individuals is calculated. This is an estimation of each individual's distance from the solution's boundaries (problem constraints or areas expected to obtain the minimum values).

- The selection of the dominating individuals takes place. The comparison between two individuals takes place in two stages. Firstly, If the individuals belong to a different front, the one with the highest-ranked front is chosen in the next iteration. If both of them belong to the same front, then their crowding distances are compared. Among them, the one with the highest crowding distance is selected, as it holds more potential to improve its classification closer to the solution boundaries.

- By default, the following steps of the algorithm opt to reject half of the population so that the next generation is again a size N.

- The new selected population P_{t+1} produces the new offspring population Q_{t+1}, and every step is repeated from the beginning.
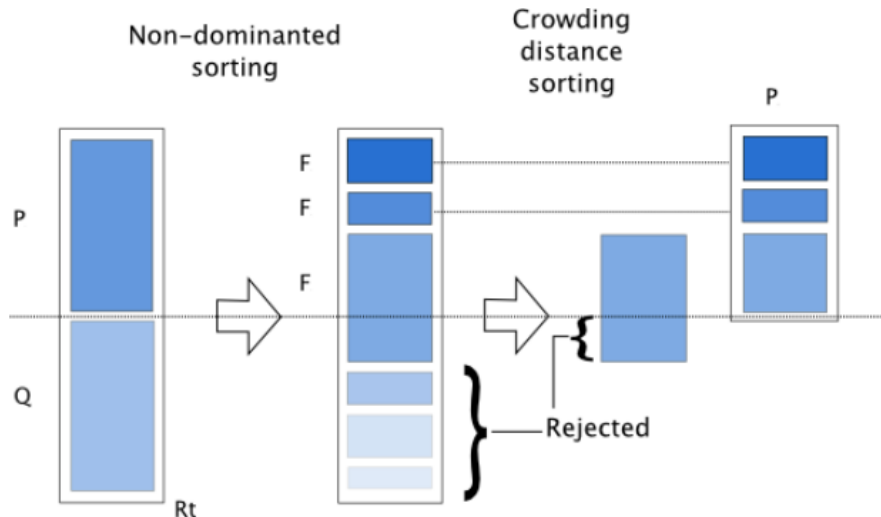
Figure 33: Schematic presentation of the NSGA-II sorting algorithm (Deb et al. [62]).

4.10 Surrogate models

With the optimization developed, a surrogate model can be implemented into the workflow in order to make the process more efficient. To implement the surrogate model, the analysis blocks inside the optimization workflow are examined in order to identify the ones that require the most computational resources. When the analysis block that needs to be replaced is identified, the inputs and the outputs connecting this block to the rest of the optimization process are identified. A representative number of input/output sets are stored in order to produce the training dataset. Finally, different surrogate models are examined to investigate which one is able to give the most reliable result.

4.10.1 Sampling

A crucial step in producing the dataset is making it as representative as possible for a given dataset size. A simple way to do that would be to divide the range of each parameter into equal ranges depending on the number of needed points. That would ensure that there are design points covering the whole design space.

A more sophisticated way to get representative samples is via Latin Hypercube Sampling (LHS, McKay et al., 1979 [63]). LHS is a very popular sampling method to obtain a representative sample. For a given number of variables, all variables are divided into the same number of equal intervals. In LHS sampling, the number of intervals for each variable is the same as the sample size, and inside each interval, only one sample point can be found.

To demonstrate the outcome of LHS, it is assumed that 10 floater samples are needed with different D_o (outer column diameter) and L_o (pontoon length), in the range [9.8,15] and [48,57], respectively. The corresponding LHS sampling is shown in Figure 34. By observing the sample, it becomes clear that for each interval of the two design parameters, there is only one point. Additionally, the LHS method can also be used for designing spaces of more than two variables.
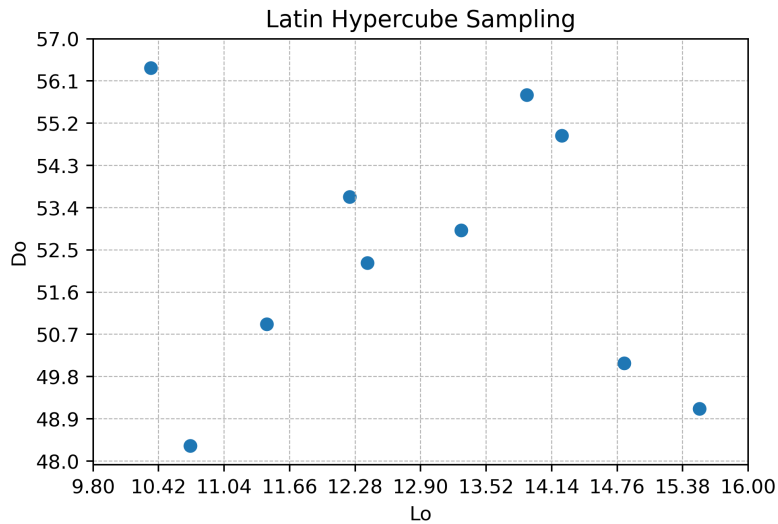
Figure 34: LHS sampling for 10 samples, regarding two variables.

With the LHS sampling applied, all the different designs corresponding to each LHS set of parameters are produced and simulated in RAFT. The outputs saved from RAFT are the ones needed to train the surrogate model. It is worth mentioning that, as shown in Section 5, the dataset needed to train the surrogate model consists of a significant number of samples. In order to simulate this large number of geometries, the High-Performance Computing (HPC) cluster of the Aerospace Department of TU Delft is utilized, providing the computational resources needed to complete every simulation within a reasonable time period.

4.10.2 Surrogate model approaches

A surrogate model is needed to replace the current optimization algorithm with one that needs significantly fewer computational resources. This can be achieved by many alternative surrogate model implementations.

As already mentioned, most of the computational resources are used to conduct the RAFT simulations. According to that observation, the simplest way to develop a surrogate model is to produce one model that, for a given set of inputs, predicts the RAFT outputs relevant to the optimization problem. Those RAFT outputs are the power quality objective function and the constraint-related outputs, being maximum surge, maximum pitch, maximum nacelle acceleration, and maximum mooring line tension and the natural frequencies of each DOF. For the rest of this thesis, this surrogate model approach is characterized as the *"End-to-End"* approach. This characterization is only made for clarification reasons, as more than one surrogate model approach is mentioned in the following sections.

If the sub-processes of RAFT are analyzed, it becomes evident that the one consuming most of the computational power provided is "PyHAMS". In more detail, the process of discretizing the geometry of the submerged floater into a mesh and solving the potential flow equations for each element of the mess requires most of the computational resources. The outcome of this sub-process is the calculation of the hydrodynamic Added mass matrix A, the hydrodynamic Damping matrix B and the Excitation matrix X. Based on that realization, an alternative surrogate model approach is to develop a model able to specifically replace the meshing and potential flow solution sub-processes of "PyHAMS". For the rest of this thesis, this model is characterized as the "PyHAMS" approach.

Schematically, the two different surrogate model approaches are illustrated in Figure 35.In the following sections, the two alternative surrogate model approaches are analyzed.
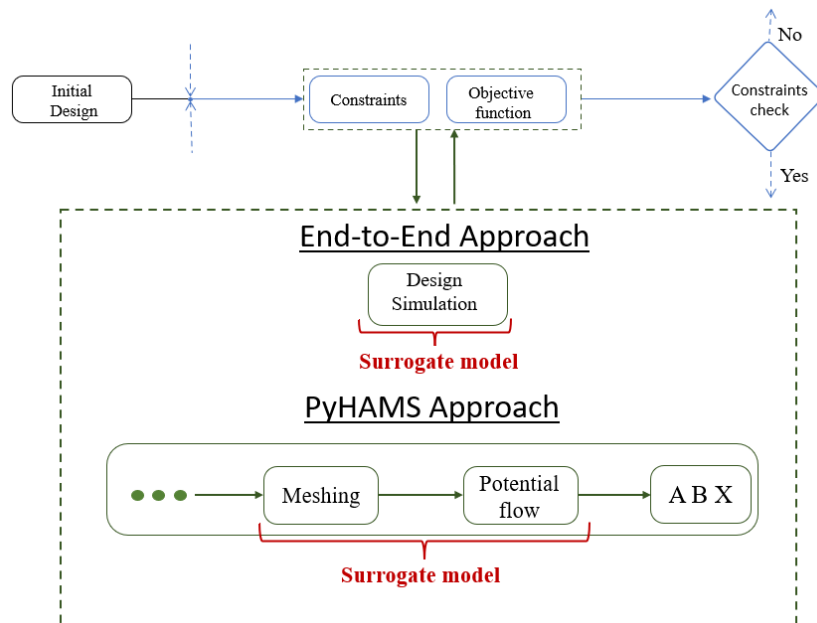
Figure 35: Surrogate model approached schematic comparison.

4.11 "End-to-End" surrogate model

As mentioned, the "End-to-End" surrogate model aims to replace RAFT entirely. Inside the optimization, RAFT is implemented for calculating the responses of the substructure and the wind turbine's power quality. Those outputs are affected by all four design parameters and the environmental conditions of each simulation. In the following sections, the inputs contained in the dataset and the surrogate model's accuracy evaluation process are analyzed.

4.11.1 Surrogate model inputs

For the "End-to-End" surrogate model, seven inputs are needed: The outer column diameter D_o, the inner-outer columns radial distance L_o, the total length of the mooring line S_t, the chain links thickness d_c, the environmental state wind speed V_{wind}, Jonswap significant wave height H_s, and peak spectral period T_p. Among the inputs, the 4 design variables are produced via LHS sampling within the design bounds. These bounds are chosen based on the suggested values of D_o and L_o analyzed in Section 4.5. For the mooring variables, the bounds chosen are within the [800, 900] m range for S_t and [0.050, 0.200] m for d_c. The reason for the choice is based on the dimensions that make practical sense for a water depth of $200m$.

Unlike the design variables, the environmental inputs can not be generated based only on LHS sampling due to the dependency between them. More specifically, there are sets of the 3 environmental variables that don't make physical sense. Due to that, the environmental design variables are generated together with the design variables by the same LHS process, but then the [V_{wind}, H_s, T_p] sets are evaluated.

The evaluation process ensures that, for a given environmental case, a randomly-selected wave height can be induced by a specific wind speed and that the wave height calculated can result in the randomly-selected wave period. Starting from the wind-induced wave height, the maximum significant wave height induced by a certain wind speed is given according to the Pierson-Moskowitz formula (Pierson and Moskowitz, 1964 [64]), being:

$$H_{s,PM} = \frac{V_{wind,19.5}^2 \cdot 0.21}{g} \tag{156}$$

where $V_{wind,19.5}$ is the average wind speed 19.5 m above the SSL, and g is the gravitational acceleration. Additionally, for a given wind speed, the corresponding wave height should be observable in the reference location. The scatter table of the wind speed versus the corresponding wave height can be found in Appendix G. Even though this additional filter leads to a surrogate model that can not be applied for every possible environmental state, it still results in a dataset able to represent most of the real-world environmental conditions. Most of the values filtered are extreme environmental conditions, which are dealt separately (as shown in the following paragraphs) in this thesis. In case a specific LHS-produced wave height is not within the V_{wind}-dependent limits, the LHS wave height is replaced with a wave height randomly selected from the available range. In addition to that, wind speeds within the [10.5, 12.5] are also excluded from the training dataset. This is related to the existence of potential errors in the calculation of σ_P, as explained in Section 4.7.2.

With the significant wave height calculated, the peak wave period value is acceptable when it is within the limits set by the DNV-GL standard [65]:

$$11,1\sqrt{H_S\left(V_{\text{hub}}\right)/g} \le T \le 14,3\sqrt{H_S\left(V_{\text{hub}}\right)/g} \tag{157}$$

where H_s is the significant wave height and V_{hub} is the wind speed on the hub height (150 m). Similarly to H_s, T_p is additionally limited based on the wave height-wave period scatter plot of the

reference location (Appendix G). According to the aforementioned limits, the resulting training dataset is shown in Figure 36a and Figure 36b.
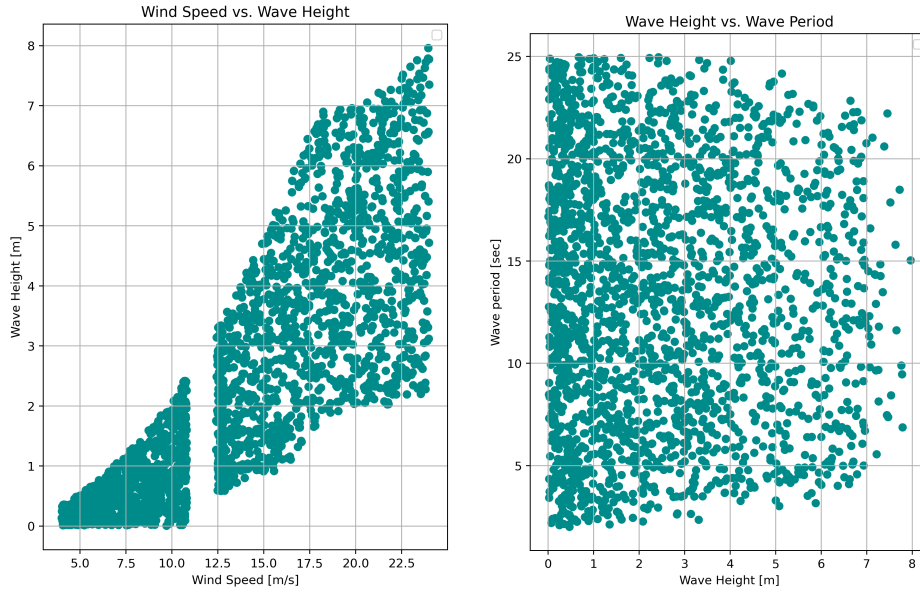
(a) Training dataset scatter plot of wind speed versus significant wave height.

(b) Training dataset scatter plot of significant wave height versus wave period.

Figure 36: Environmental conditions samples of the "End-to-End" training dataset.

Regarding the extreme environmental case, a training dataset is specifically produced for this one. On one hand, the extreme environmental condition defined in this thesis does not comply with the limits between V_{wind}, H_s and T_p, as the rated wind speed is combined with an extreme wave. On the other hand, if both the normal and extreme environmental conditions were to be included in the same training dataset, a significant part of the dataset would include samples that are not needed for either one of the two environmental states (normal or extreme). That way, a much larger dataset would be needed without any additional benefits to the optimization problem. For the extreme dataset used in this thesis, the environmental conditions are not inputs in the surrogate model, as they are kept constant, and only the design variables are the ones included in the training dataset. Furthermore, the power quality output is not included in the dataset, as the extreme environmental condition is not assumed to happen on a specific frequency (hence it has a practical probability of occurrence of zero).

Finally, it is worth mentioning that each environmental condition defined in RAFT also includes the environmental turbulence intensity. For this thesis, the turbulence intensity is not included in the training dataset, as it is derived directly from the wind speed, as analyzed in Section 4.2 for both the normal and extreme environmental conditions.

4.11.2 Surrogate model accuracy

With the sets of input parameters defined for the "End-to-End" approach, the different sets are simulated in RAFT. After each simulation is completed, the set of inputs and its corresponding outputs of interest are stored. The outputs of interest are the PQ, the maximum pitch angle, the maximum surge displacement, the maximum nacelle acceleration, the maximum mooring line tension and the natural frequencies of the substructure (with the exception of the extreme environmental case).

When every simulation is completed, and the results are stored, the surrogate model training dataset is completed. This training dataset is used for training different surrogate models. The surrogate models tested for this thesis are Linear Regression (LR), Decision Trees (DT), Random

Forest (RF), Extreme Gradient Boosting (XGBoost), and Artificial Neural Networks (ANN). The reason for the choice is based on the prediction requirements. Specifically, the surrogate model needed should be able to make a prediction for a combination of inputs. In other words, a multi-variable surrogate model is needed. This is why most of these models are Decision Tree based, as, together with ANNs, they are expected to perform well for such a dataset. At the same time, these models are expected to be able to predict output behaviours that the LR model is not able to predict accurately. Finally, The performance of the different surrogate models is evaluated and compared.

The performance of each model is evaluated according to the K-Fold Validation (Hastie, 2009 [66]). In this method, the whole training dataset is split into a (user-specified) number of subsets that do not overlap with each other. Those subsets are called Folds.

For each Fold of the dataset, an evaluation process takes place. This process uses the rest of the Folds (apart from the evaluated one) to train the surrogate model. This surrogate model is used to predict the output for the set of inputs contained in the unused Fold. Finally, the error between the real and the predicted output is evaluated and the next Fold is evaluated.

There are many ways to evaluate the accuracy of the fit between the prediction and the real output. In this thesis, the accuracy is quantified based on the coefficient of determination, also known as the R^2 *score* (Barrett, 1974 [67]). This coefficient, ranging from 0 to 1, is widely used in surrogate model fitting evaluation. The closer the value is to 1, the better the surrogate model is able to predict the output of the system and also explain its variability. The main advantage of this coefficient is that it evaluates the quality of the fit based on the deviation of the real dataset. This gives a better insight into the accuracy of the surrogate fit, as it demonstrates the significance of the error between the prediction and reality. For instance, if the dataset deviates from its mean value within the $\pm 10\%$ range, a 0.1% error from the surrogate can be considered insignificant (depending on the case). On the contrary, if the dataset deviates within the $\pm 0.5\%$ range, it is most likely that the same surrogate model error is significant.

Mathematically, the R^2 score is given by:

$$R^2 = 1 - \frac{SSE}{SST} \tag{158}$$

where the SST and SSE are the total variability or dispersion of the target (real) and the predicted (surrogate) output, respectively, given by:

$$SS = \sum_{i=1}^{n}(y_i - \bar{y})^2 \tag{159}$$

In the case of SST, y represents the pre-calculated output of the model contained in the training dataset. Similarly, when SSE is calculated, y is the predicted output from the surrogate model.

With the R^2 score calculated for one of the training dataset Folds, the next fold is evaluated. After every Fold is evaluated, the performance indicator of each model is the mean value of each R^2 score calculated for every Fold, $\bar{R}^2$. The K-fold validation process described above is also illustrated in Figure 37, for the case of 9 Folds .
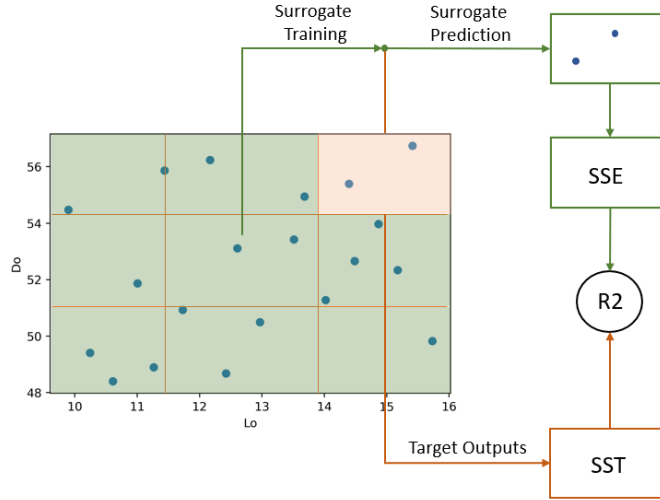
Figure 37: K-Fold validation process schematic, for a process of 9 Folds.

Finally, it is worth mentioning that, in the "End-to-End" surrogate model, there is one surrogate model predicting each one of the outputs. Hence, the algorithm predicting the outputs of the system utilizes 12 surrogate models (1 objective output, 4 RAFT-derived constraints, and 6 natural frequencies). Nevertheless, the computational time needed to train all the models and produce each output from a different surrogate model is insignificant compared to the optimization computational time. Also, even though utilizing different models for each output is feasible, it is not tested in this thesis. However, it is expected that there are insignificant advantages, as it is later demonstrated that all surrogate models perform similarly in most outputs, with the exception of some specific ones, which determine the model used.

4.12 "PyHAMS" surrogate model

In order to develop the surrogate model for replacing the process of producing a mesh and solving the potential flow theory, a similar philosophy as the one followed in Section 4.11 is followed. In this case, though, the inputs required are only those that affect the properties of the floater, being D_o and L_o. Regarding the prediction output, "PyHAMS" calculates the hydrodynamic Added Mass, Damping and Excitation matrices, A, B and X, respectively. Each one of these matrices has a size of 6x6x80 (1x6x80 for X), with the 6x6 dimension corresponding to the floater's DOFs, and 80 to the size of the discretized frequency vector.

However, not all the elements of these matrices need to be predicted. A significant amount of elements in A, B, and X are (practically) zero. Even though the floater represents a 6 DOF system, the wind-wave direction with respect to the floater's orientation (Figure 38) makes the problem x-axis-symmetric. Hence, the sway and yaw responses are (practically) zero. Additionally, the generator torque can introduce a roll excitation to the system, but its amplitude is expected to be insignificant. For that purpose, all the elements related to these DOFs are zero and don't need to be predicted. It is worth noting, though, that the system is still described as a 6 DOF, as RAFT's current version can not solve different DOF systems.

Regarding the investigation of the accuracy of this model, the same models used in Section 4.11 are utilized, with the exception of ANNs. Even though the ANNs models hold the potential to predict each one of these vectors, the complexity of setting up a model of significant accuracy increases dramatically, due to a large number of outputs, compared to the inputs (80 related outputs from 2 inputs). Even though an alternative approach to set up a model is to predict each one of the vector elements of each vector, this approach is expected to decrease significantly the efficiency of the surrogate model. Nevertheless, it is expected that most of the remaining models will achieve reliable prediction accuracy.

It is also worth mentioning that, regardless of the accuracy of the "PyHAMS" approach in predicting the hydrodynamic matrices, it is important to investigate the accuracy achieved in predicting the final outputs needed for the optimization framework. Due to the fact that the hydrodynamic matrices are not the output directly utilized by the optimizer, apart from evaluating the accuracy of the surrogate model, the accuracy of implementing the surrogate model inside the RAFT workflow should also be examined. This is also important as it is vital to examine if the final output of RAFT is affected from the potential inaccuracies of the surrogate model.
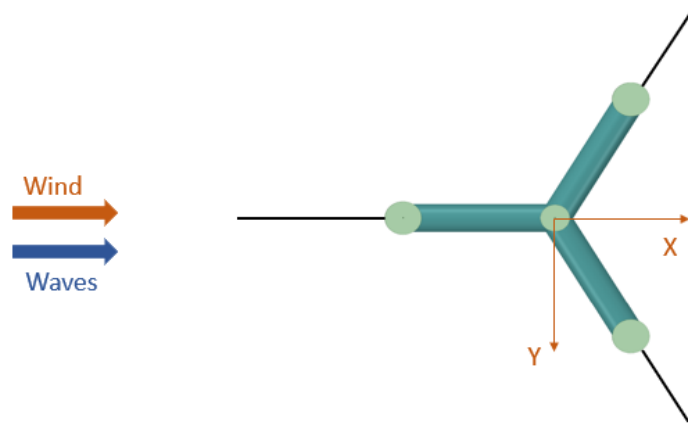
Figure 38: Wind/waves direction, with respect to the orientation of the substructure.

5 Results

This section summarises the results obtained in this thesis. Firstly, the two different surrogate modelling approaches, "PyHAMS" and "End-to-End", are assessed based on their accuracy for a number of different surrogate models. Then, the Pareto front population is analyzed, based on the results obtained from a RAFT-based optimization. The same optimization process is conducted via the "End-to-End" and "PyHAMS"-based optimization algorithms, comparing the optimization outcome with the results derived from RAFT. Finally, the computational efficiency achieved by each model is presented.

5.1 PyHAMS approach

5.1.1 Surrogate models comparison

As already mentioned, the "PyHAMS" approach predicts the hydrodynamic added mass (A), waves' radiation damping (B), and waves' excitation (X) matrices of a floater design for a given D_o and L_o value. All the elements of the matrices are frequency dependent. Therefor, the predicted output for each property of each DOF is a 1x80 vector, corresponding to the values of one specific matrix element for a given frequency.

Four different surrogate models have been assessed for the "PyHAMS" approach. These are: Linear Regression (LR), Decision Trees (DT), Random Forest (RF), and Extreme Gradient Boosting (XGBoost). For each one of these models, the prediction accuracy of the diagonal, frequency-dependent elements of all A, B and X matrices is evaluated. For each element, the R^2 score for every frequency is obtained based on a 10-Fold validation. Finally, in order to compare the models, the average value of the 80 R^2 scores (one for each frequency of the frequency vector) is used. The dataset used to compare the surrogate models features 600 samples. The reason for the choice is that this dataset is significantly larger than what is needed in order to obtain reliable fits, with an R^2 score ≥ 0.9 for at least one of the models. The performance of this model with respect to the variation of the dataset size is discussed in detail in Section 5.1.2.

Figure 39 compares the fit accuracy of the four different models, based on the 10-Fold validation R^2 score. Further details about the hyperparameters utilized for each surrogate model can be found in Appendix J. Evidently, the most reliable model is the XGBoost, which outperforms the rest of the models. Compared to the DT and RF models, this is a natural outcome, as the XGBoost model is an enhanced version of the RF. Similarly, the RF model is a collection of Decision Trees. This is the reason why, for all outputs, the Random Forest method outperforms DT.
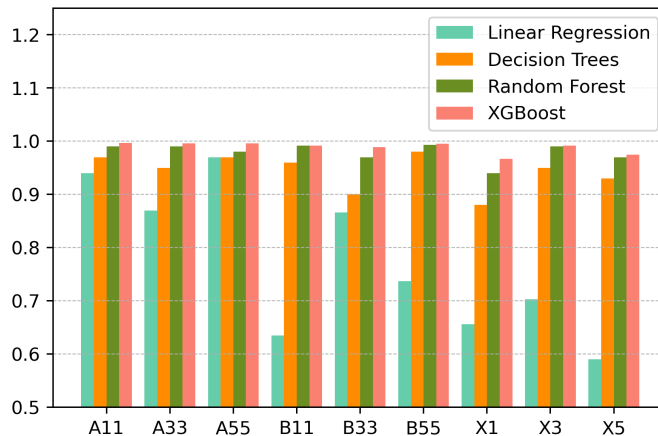
Figure 39: Surrogate models R^2 score comparison, for the "PyHAMS" approach.

The Linear Regression model performs poorly in predicting the damping and excitation properties of the floater, indicating that the waves' radiation damping and waves' excitation properties can not be accurately estimated by linear functions. Instead, LR performs better in predicting the added mass elements, implying that a linear function of D_o and L_o can predict added mass with a certain accuracy.

In terms of training time, LR needs the least time to train the model for the 600 samples dataset, followed by DT. Both RF and XGBoost need similar computational times to train the models. However, it is observed that the training time is insignificant compared to the total time required to conduct the "PyHAMS"-based optimization process. In fact, training the surrogate model is a process that requires approximately 3 minutes, with minor variations depending on the model. Thus, training time was not considered a relevant metric to selecting the surrogate modelling technique adopted in this thesis.

5.1.2 Training dataset size

One important aspect of surrogate models is related to the size of the training dataset provided to the model. This is important as the computational resources needed to produce the training dataset for the "PyHAMS" surrogate model should be less than what is needed to conduct the optimization based on the frequency domain analysis. Otherwise, there is no practical advantage in utilizing a surrogate-based optimization instead of the FD-based one.

In order to investigate the effect of the dataset size on the surrogate model's accuracy, training datasets of different sizes are produced, and the (vector) average R^2 score is obtained for every diagonal element of the predicted matrices. Due to the high computational resources needed to produce the training datasets, it is not possible to produce an extensive number of different LHS-based datasets. Thus, especially for datasets with a significant number of samples, the number of different datasets examined is limited. Nonetheless, the datasets tested provide the information needed to conclude a dataset size. Figure 40 demonstrates the average R^2 score increase after a 10-Fold validation for a given LHS dataset size. Evidently, the model's accuracy increases with the number of samples.

In this thesis, a dataset of 350 samples is used for the "PyHAMS" surrogate. This choice is based on three major factors. Firstly, all the model's outputs achieve an R^2 score higher than 0.95, indicating a very good fit. Secondly, it is clear that, for datasets larger than 350 samples, the accuracy increase is insignificant. Finally, the time required to produce the 350 samples dataset is reasonable. To demonstrate, the computer used in this thesis (see Section 5.5) produces this dataset in approximately 23 hrs. For the optimization problem of this thesis, this timeframe is not considered excessive. If the production time needs to be reduced, smaller datasets can also be utilized without significantly reducing the R^2 score. For instance, the 250 samples dataset provides similar R^2 score values for almost every output while reducing the production time by approximately 5 hrs. Choosing the dataset size depends on the efficiency and accuracy aiming to achieve. For this thesis, the trade-off of increasing the dataset production time is not considered significant.
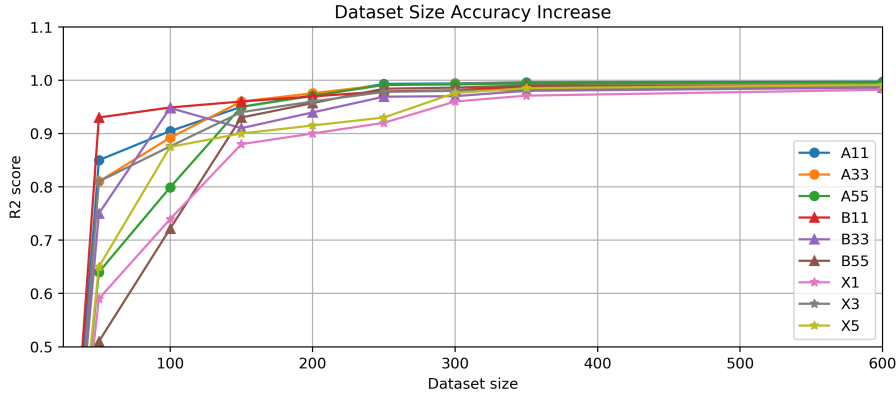
Figure 40: Accuracy of the "PyHAMS" approach surrogate model outputs as a function of the dataset size.

5.1.3 XGBoost accuracy

As demonstrated in the previous section, XGBoost performs better than the other three surrogate modelling techniques evaluated. Nonetheless, the surrogate model's capability of predicting the PyHAMS outputs remains to be determined. In fact, the R^2 scores presented in Section 5.1.1 are frequency-averaged, while it is important to analyze the accuracy of the model for each frequency. Furthermore, the sensitivity of the key RAFT outputs, given the potential error of the surrogate model, also needs to be determined. Since the "PyHAMS" approach replaces a specific analysis block inside RAFT, it should be verified that the key optimization-related outputs needed from RAFT are also accurately predicted. Even for a very well-predicted output from the surrogate model, the RAFT output might be very sensitive to small mismatches between the PyHAMS and surrogate model derived result.

A 10-Fold validation is conducted for each frequency of each vector mentioned in Section 5.1.1, being the Added mass, waves' radiation damping, and waves' excitation in surge, heave and pitch. In this analysis, a training dataset of 350 samples is utilized. The results of this analysis are shown in Figure 41 to Figure 43. It can be noticed that the XGBoost method ensures a very accurate prediction (R^2 score ≥ 0.92) for almost every frequency of the nine vectors. This indicates that the model is able to accurately predict the same elements in the matrices for every frequency. However, there are some limited frequency ranges with lower R^2 scores. This reduced accuracy is observed in the pitch damping (B55), surge excitation (X1) and pitch excitation (X3) vectors. However, in all these cases, the quantities to be predicted are small compared to the remaining frequencies. The value of B55, X1 and X5 in the low-frequency range is practically zero. Due to that, the RAFT-derived value is affected by insignificant random errors (for instance, by errors in the potential flow solution). As a result, the surrogate model indicates reduced accuracy in predicting outputs that are subject to randomness. Nevertheless, due to the insignificance of B55, X1 and X5 in the low-frequency range, the error does not affect the predicted output.
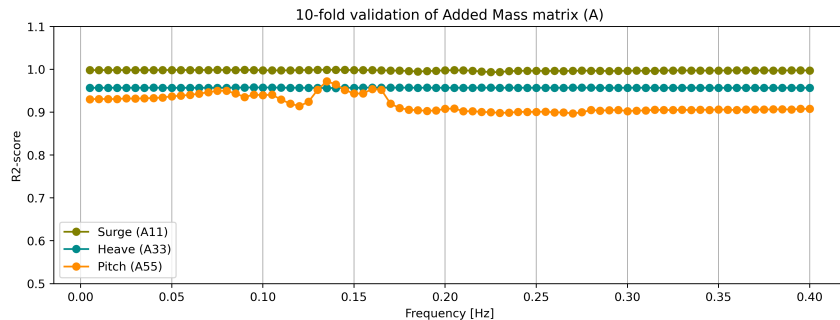
Figure 41: 10-Fold validation for the added mass matrix diagonal elements

68
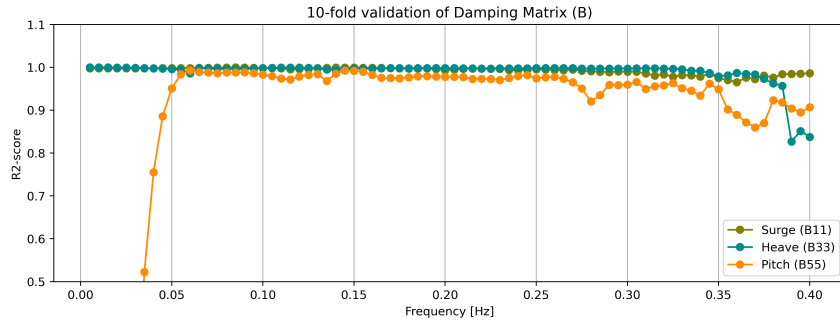
Figure 42: 10-Fold validation for the waves' radiation damping matrix diagonal elements
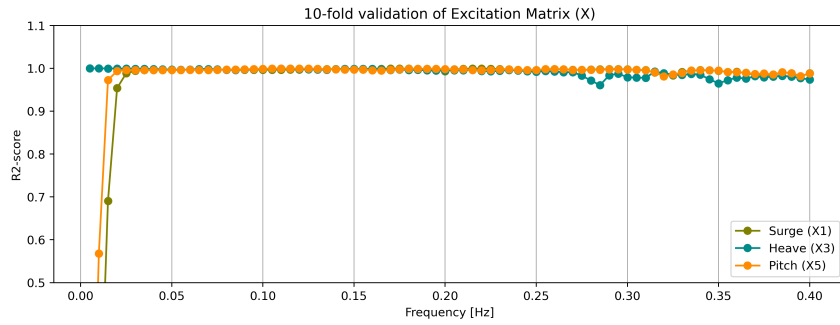
Figure 43: 10-Fold validation for the waves' excitation matrix diagonal elements

To demonstrate the high accuracy achieved by the "PyHAMS" approach surrogate model, Figure 44 presents the diagonal element vectors of the A, B, and X matrices, calculated by RAFT and the surrogate model. The presented results are calculated for an arbitrary design, not included in the training dataset. The same comparison was carried out for several designs with identical results. From Figure 44, it is clear that the surrogate model is very capable of accurately predicting the hydrodynamic vectors. In almost every scenario, the waves' radiation damping and waves' excitation predictions are almost identical. Additionally, it can be noticed that, in the regions where the R^2 scores are lower, the value of the predicted quantity is close to zero. Thus, regardless of the lower R^2 score value, the prediction error does not affect the output. Regarding the added mass vectors, it is noticeable that the prediction of A_{33} and A_{55} deviates slightly from the real output. Nevertheless, the prediction can be considered accurate.
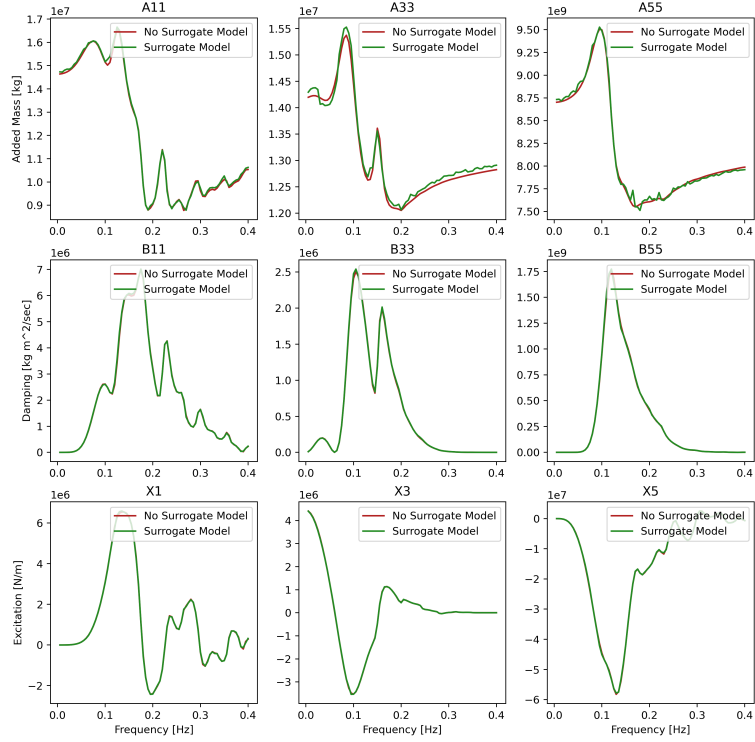
Figure 44: Comparison between the diagonal elements output from RAFT and the "PyHAMS" surrogate model approach.

It becomes clear that the surrogate model accurately predicts the frequency-dependent hydrodynamic properties of the semi-submersible floater. Regardless of the potential flow solution prediction, what is needed from RAFT in the optimization workflow is the objective function and the constraint-related outputs. To investigate the accuracy obtained in these outputs, 300 different design configurations are simulated, both by the original RAFT algorithm and the surrogate-based one. These designs keep the same St and dc, but their D_o and L_o vary within the optimization's bound for those two variables. Based on the prediction and the actual value of the RAFT-derived variables calculated, the R^2 score is obtained for each output and indicated in Table 10.

	Power Quality	Surge Max	Pitch Max	Mooring Tension Max
R^2 score	0.99	0.97	0.99	0.95

Table 10: R^2 scores of the RAFT-derived outputs.

Even though the R^2 score is an established accuracy index, to further demonstrate the accuracy of the surrogate model, the deviation error between the RAFT and surrogate model outputs is shown in the contour map of Figure 45. This error is defined as:

$$\epsilon = \frac{Y_{i,RAFT} - Y_{i,SM}}{\sigma_{sample}} \tag{160}$$

With $Y_{i,RAFT}$ and $Y_{i,SM}$ being the output of sample i based on the original RAFT and the surrogate algorithm, respectively. This error representation is chosen in order to demonstrate the significance of the deviation between two points compared to the typical standard deviation of the sample. Since the surrogate model is developed for an optimization algorithm, it is important to

ensure that, for a given error between two points, the prediction's error is not significant compared to the standard deviation found in the design space. The contour plots of the compared outputs values inside the design space, as calculated by RAFT, can be found in Appendix I.

By observing Figure 45, it can be seen that, for most of the design space, the error between the two methods is insignificant (less than 0.2σ). Also, since the error is always positive, the "PyHAMS" approach consistently underestimates the outputs demonstrated ($Y_{i,RAFT} > Y_{i,SM}$, Equation 160). However, there are specific areas inside the design space where the error is large. This error increase is related to two phenomena. Firstly, for the lower boundaries of the design space, and especially close to the lower boundary of L_o, the mooring lines' slackness is reduced (they become tauter) due to the reduction of the floater size. This slackness reduction makes the 840-fix length of the mooring lines fully suspended during the simulations, affecting their expected behaviour. An example of this observation can be seen in the bottom left corner of the maximum surge response. Due to the very low extension margin of the mooring lines, the maximum surge response is almost zero. However, these design configurations are excluded from the feasible design space, as the vertical load on the anchor constraint is violated. Considering that constraint, the poorer performance of the surrogate model in this sub-region of the design space does not affect the optimization. Secondly, at the boundaries of the design space, there are fewer training samples available in comparison with the rest of the training dataset. Therefor, it is expected that the surrogate model's performance is reduced in this area.

Also, regarding the maximum surge and mooring line tension errors, the average error observed is insignificant on average. More importantly, though, those two parameters are mainly dependent on the mooring characteristics of the substructure and not on the floater ones. The insignificance of the floater design on the maximum surge and mooring tension can be observed in the RAFT-derived contour plots (Appendix I), by comparing the effect of configurations in the floater and the mooring lines' design. As shown, the standard deviation for floater configurations is insignificant, compared to modifications on the mooring lines. Thus, regardless of the error introduced by the surrogate model in those two outputs, their output values are mainly determined by the quasi-static problem's solution, which is not significantly affected by the hydrodynamic matrices computed by the "PyHAMS" approach.
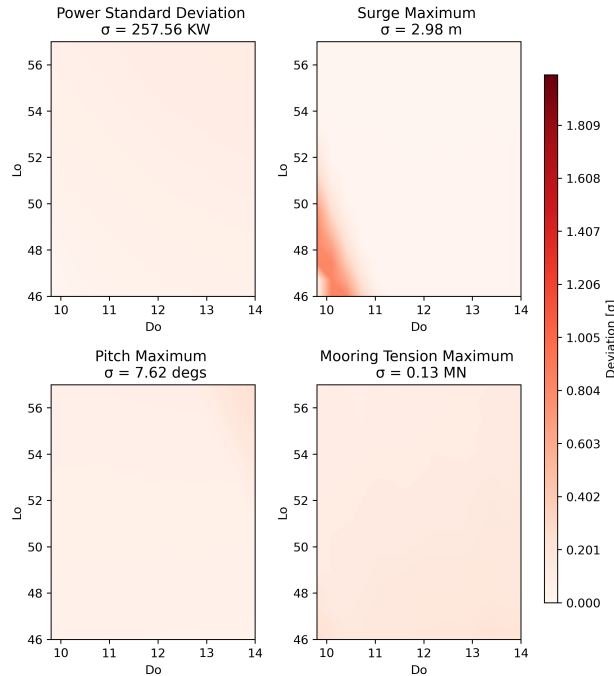

Figure 45: Error between RAFT and surrogate model-based outputs, with respect to the dataset's standard deviation.

5.2   "End-to-End" approach

With the implementation of the "End-to-End" approach, RAFT is entirely replaced inside the optimization process by two surrogate models. These models are able to predict the key parameters produced by RAFT to be evaluated against the optimization problem. One model predicts the outputs related to the operational environmental conditions, and another one the outputs related to the extreme load case. For the operational environmental cases, the inputs of the surrogate model used are the 4 design variables (D_o, L_o, S_t, d_c) and the 3 environmental inputs needed to define an environmental case (V_{wind}, H_s, T_p). The predicted outputs are the power standard deviation (power quality), maximum surge response, maximum pitch response, maximum nacelle acceleration and maximum mooring line tension. For the extreme environmental case surrogate model, the environmental condition is unique. Thus, the extreme condition model's inputs are the 4 design variables. Also, for the extreme condition model, power quality is not predicted. The extreme event's annual probability of occurrence is practically zero. Thus, the extreme condition does not contribute to the power quality calculated. The following paragraphs focus on the surrogate model utilized for the operational environmental case, followed by an analysis of the extreme condition model.

5.2.1   Surrogate models comparison

Similar to the actions taken on the "PyHAMS" approach, five different surrogate models are compared to identify the most suitable approach. The models examined are: Linear Regression (LR), Decision Trees (DT), Random Forest (RF), Extreme Gradient Boosting (XGBoost) and Artificial Neural Networks (ANN). For each model, the prediction accuracy of all the outputs needed is evaluated based on the R^2 score of a 10-Fold validation. Further details about the hyperparameters utilized for each surrogate model can be found in Appendix J. This comparison is shown in Figure 46. The dataset used consists of 1800 samples, as it is observed that at least one of the models reaches a 10-Fold validation R^2 score higher than 0.9.

By observing Figure 46, it becomes clear that the most suitable surrogate model is again the XGBoost model, as it outperforms every other surrogate model in every output. The XGB model shows a higher performance than the RF one, which outperforms DT. This observation is in line with the one made about the "PyHAMS" approach models' comparison (Section 5.1.1). The higher fitting accuracy of the LR model in the maximum surge output indicates a significant linear dependence between it and the model's inputs. Finally, regarding the ANN model, several different structures are examined. These structures vary in the number of layers and number of neurons utilized. However, for the best combination of the above-mentioned factors, the accuracy of the model is comparable with the DT model. Even in this case, the accuracy of the standard deviation of power and the maximum mooring lines' tension is significantly less. A strategy that could potentially increase the accuracy of the ANN model is to build a different structure for each predicted output. In this thesis, however, this option is not examined.

Even though the R^2 score is a reliable indicator to obtain an overview of the model's accuracy, it is important to investigate the performance of the model further. Especially for the "End-to-End" model, the design space defined by the dataset is large because it involves seven inputs. Due to that size, the training dataset's samples are not as densely arranged as compared to the "PyHAMS" approach.
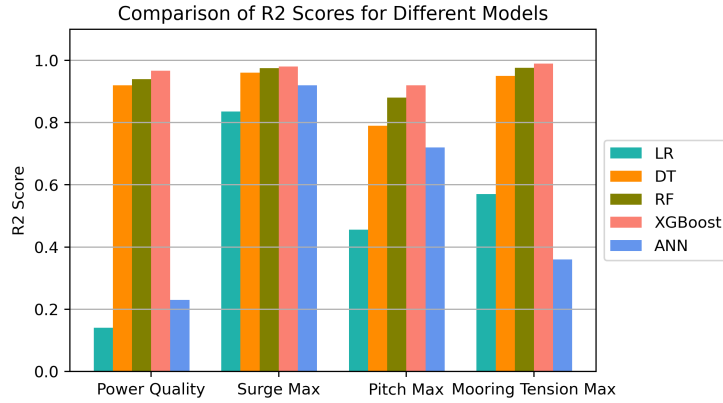
Figure 46: Surrogate models R^2 score comparison, for the "End-to-End" approach.

5.2.2 Training dataset size

In order to investigate the effect of the dataset size on the surrogate model's accuracy, training datasets of different sizes are produced, and the 10-Fold validation R^2 score is obtained for every output of the model. Due to the high computational resources needed to produce the training datasets, it is not possible to produce an extensive number of datasets. Thus, only a small number of different dataset sizes have been tested. Also, the largest training dataset tested consists of 1800 samples. This dataset's size is already very extensive, and the computational time needed to be produced is dramatically higher compared to the optimization time needed to conduct an optimization based on RAFT.

Figure 47 shows the average R^2 score increase after a 10-Fold validation for a given LHS dataset size. Evidently, the model's accuracy increases with the number of samples. Depending on the accuracy required by the surrogate model, the dataset size chosen can differ. Furthermore, it is clear that even for a dataset of 1800 samples, some of the outputs still show an increasing trend. This is in contrast with what is observed in the "PyHAMS" approach (Section 5.1.2), where after 350 samples, there is no significant increasing trend. Among the "End-to-End" approach outputs, the only one that does not show significant increasing trends is the maximum surge response, the accuracy of which is practically constant after the 1200 samples dataset.

As already explained, this thesis utilizes a dataset of 1800 samples. This choice is based on two major factors. Firstly, all the model's outputs achieve an R^2 score higher than 0.90, indicating a high prediction accuracy. Secondly, generating large datasets is computationally expensive, making the dataset's extension difficult to justify. Especially considering the fact that this surrogate modelling approach requires another extreme case training dataset, the computational time required for both datasets is immense.
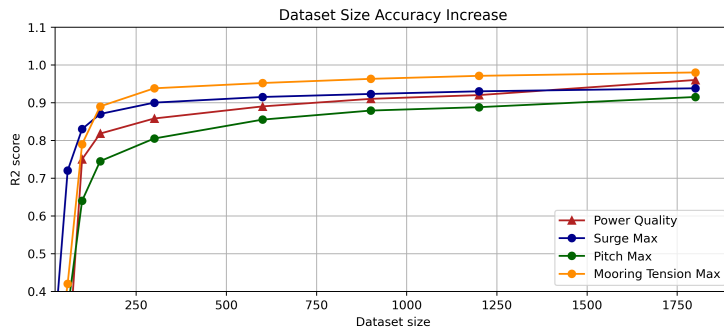
Figure 47: Accuracy of the operational conditions "End-to-End" approach surrogate model outputs, as a function of the dataset size.

5.2.3 XGBoost accuracy

In order to further investigate the accuracy achieved by the XGBoost model, the outputs computed by RAFT and the "End-to-End" model are compared for 600 different designs. The first 300 designs correspond to variations of the floater design variables (D_o and L_o), for constant mooring characteristics. The remaining 300 designs correspond to the configurations of the mooring lines design variables (S_t and d_c), for constant floater characteristics. Also, the operational environmental conditions of the simulations are the reference ones mentioned in Table 9. That way, the accuracy of the model is tested for a plethora of different environmental inputs.

Figure 48 illustrates the error, as defined in Section 5.1.3, between the outputs from RAFT and the "End-to-End" approach. This error is with respect to the floater's design variables. Although the R^2 score is higher than 0.9, it becomes clear that the deviation error is significantly higher compared to the "PyHAMS" approach, shown in Section 5.1.3. Additionally, unlike the "PyHAMS" approach, which consistently underestimates the outputs, the "End-to-End" approach predictions do not follow any specific underestimation or overestimation trend.

Starting from the power standard deviation output, σ_P, for the majority of the design space, the prediction error is moderate, with a value less or equal to $\pm 0.4\sigma$. The only area where the error observed is higher is the left boundary of the design space, where the deviation error can affect the optimization result. Furthermore, there is not a clear trend in the error observed, as there are both positive and negative error regions. There error regions are also scattered inside the design space, without any specific location trends.

Figure 48: Error between RAFT and surrogate model-based outputs, with respect to the dataset's standard deviation (floater characteristics variation).

On the contrary, the deviation error of the maximum pitch is quite significant. On one hand, prediction errors higher than 0.4σ cover an extensive part of the design space. On the other hand, there is a switch between an underestimation error area to an overestimation one, which can potentially affect the morphology of the contour lines close to the error switch area. Even though this prediction error can potentially affect the optimization outcome, among the reference environmental conditions, the most decisive one on the maximum pitch angle is the extreme environmental condition, which is handled by the extreme condition surrogate model.

Finally, the maximum surge and maximum mooring lines' tension output errors are very significant, with average error values higher than 1.5σ. Even though those predictions can not be considered accurate, the standard deviations observed in those two floater-dependent outputs are insignificant compared to the standard deviation observed due to modifications on the mooring lines (Figure 49). For that reason, the potential that these output errors affect the optimization is considered minor.

Regarding the model's prediction accuracy for modifications on the mooring lines characteristics, Figure 49 demonstrates the error of the above-mentioned outputs, with respect to the mooring lines' characteristics. Starting from the two mooring-dependent outputs, the error observed in the maximum mooring lines tension is insignificant (less than 0.2σ) for the majority of the examined design space. On the contrary, the maximum surge response error observed is significant in certain areas. In detail, close to the right boundary of the design space, the error is higher than 1σ. This is also the region where the maximum surge constraint is expected to activate. However, as discussed before, it is also important to take into account that the maximum surge is expected to be driven by the extreme environmental case, which is not computed from this surrogate model.

Unlike maximum surge and mooring lines tension, the maximum pitch angle's prediction error is significant. Nevertheless, the standard deviation observed due to mooring modifications is minor compared to the one observed in Figure 48. For power standard deviation, the prediction contains a moderate error (mainly in the range of 0.4σ). This positive error also indicates an underestimation of σ_P. However, at the left boundary of the design space ($S_t \leq 820$), the prediction error is significant (frequently higher than 1σ). In the area of high error, though, it is expected that the low S_t value will violate the anchor vertical loads constraint.
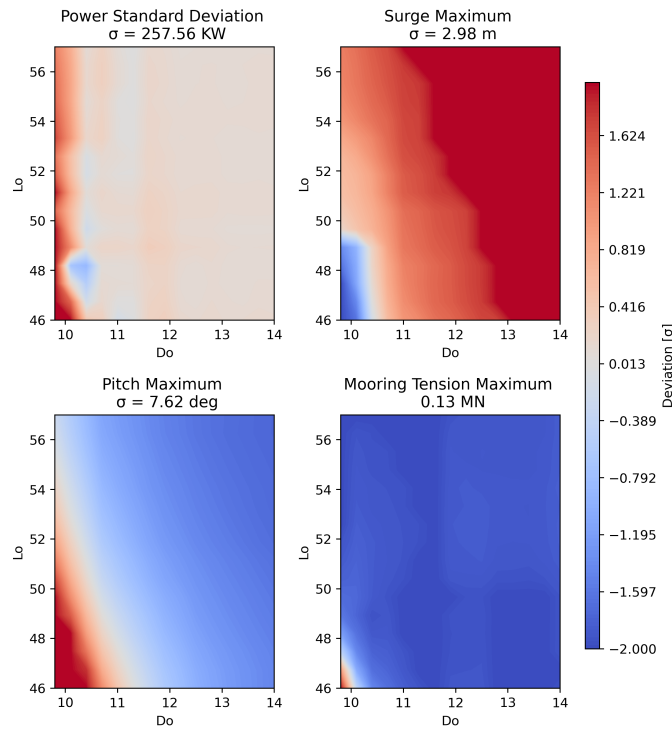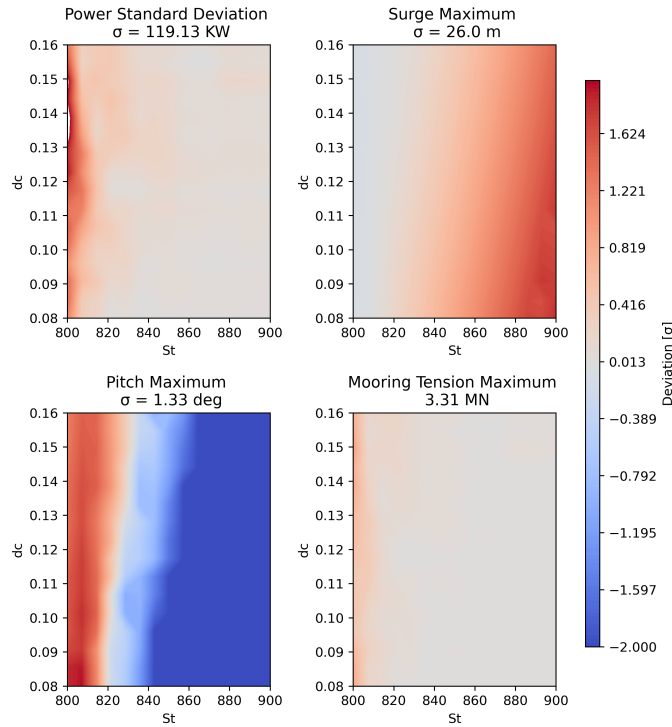

Figure 49: Error between RAFT and surrogate model-based outputs, with respect to the dataset's standard deviation (mooring characteristics variation).

5.2.4   Extreme environmental case

In the "End-to-End" approach, dedicated surrogate models calculate the responses of the sub-structure for the operational and extreme environmental conditions. The reasoning behind that decision is that there is a wide gap between the environmental characteristics of the operational and extreme conditions. Those characteristics are the wave height, H_s, wave period, T_p, and the turbulence intensity, which is calculated according to an extreme turbulence function (see Section 4.2). If that gap is included in order to train a surrogate model for both cases, the size of the dataset should be significantly larger than the one utilized in the current approach. Furthermore, by keeping the extreme environmental conditions constant, the inputs needed in this surrogate model are 4, implying that a smaller dataset can provide reliable accuracy.

In order to identify the most suitable surrogate model for this application, the same process followed in Section 5.1.1 is conducted. Specifically, the LR, DT, RF, XGBoost and ANN models are used to train the extreme case surrogate. For each model, a 10-Fold validation is conducted in order to obtain the R^2 score of each output. Further details about the hyperparameters utilized for each surrogate model can be found in Appendix J. According to the results of the 10-Fold validation method, the model achieving the highest accuracy is the XGBoost surrogate model, with the R^2 score of each output shown in Figure 50
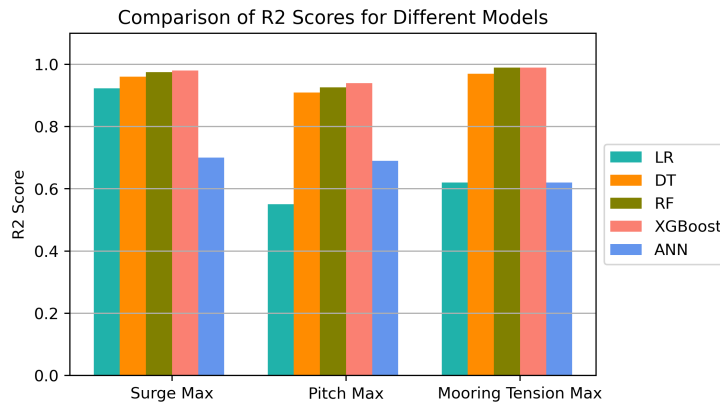
Figure 50: Surrogate models R^2 score comparison for the extreme environmental condition of the End-to-End approach.

Similarly to the process followed in the other two surrogate models, different dataset sizes have been compared in order to evaluate the accuracy that can be achieved by the model. For this case, the smallest dataset able to reach a 10-Fold validation score of 0.9 in every output is chosen. The reasoning behind not exploring the potential of larger datasets is that for the "End-to-End" approach, the operational conditions dataset is already large. Hence, avoiding increasing the total production time of both datasets is important. The dataset that meets these conditions consists of 600 samples, as shown in Figure 51. Figure 51 also shows that the accuracy of the model can be further improved, if needed, by increasing the number of samples. In fact, even for the 600 samples dataset, there is still a moderately increasing trend, with emphasis on the maximum pitch output.
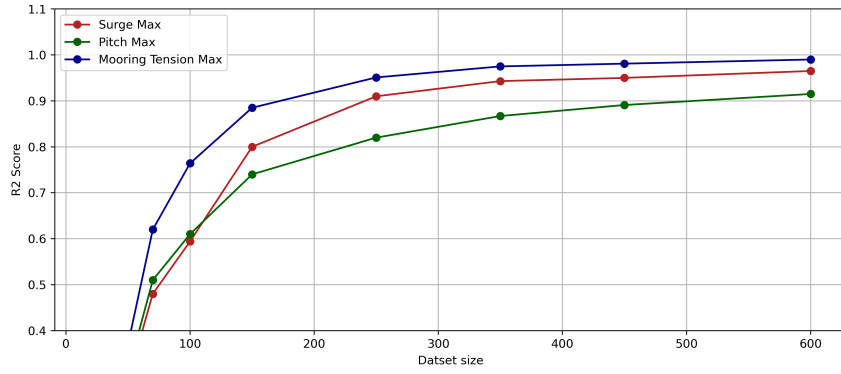
Figure 51: Accuracy of the extreme condition "End-to-End" approach surrogate model outputs, as a function of the dataset size.

Unlike the other two surrogate models analyzed in this thesis, this model's accuracy is evaluated based on its prediction of the 3 constraints deriving from the model's outputs. These constraints are the maximum surge, maximum pitch, and maximum vertical tension at the touchdown constraint. The extreme case model is not applied for calculating the power quality of a specific design, and neither can it be used for a variety of different conditions. Due to its lack of versatility, evaluating the accuracy of the model at predicting the limits of the constraints can give a better insight to the "End-to-End"-based optimization.

Figure 52 compares the maximum surge, pitch and anchor stability constraints, as calculated by RAFT and the extreme environmental case surrogate model. The solid lines indicate the constraint limit, as calculated by the "End-to-End" approach, and the dashed line corresponds to the same limit, as calculated by RAFT. As shown, the accuracy of the surrogate model is in line with the R^2 score reported in Figure 50. Among the three constraints, the maximum surge and anchor's stability constraints are accurately predicted, as the surrogate model's constraint matched closely the RAFT output. Instead, for the pitch constraint, although the prediction of the surrogate follows the expected trend, the error is larger.
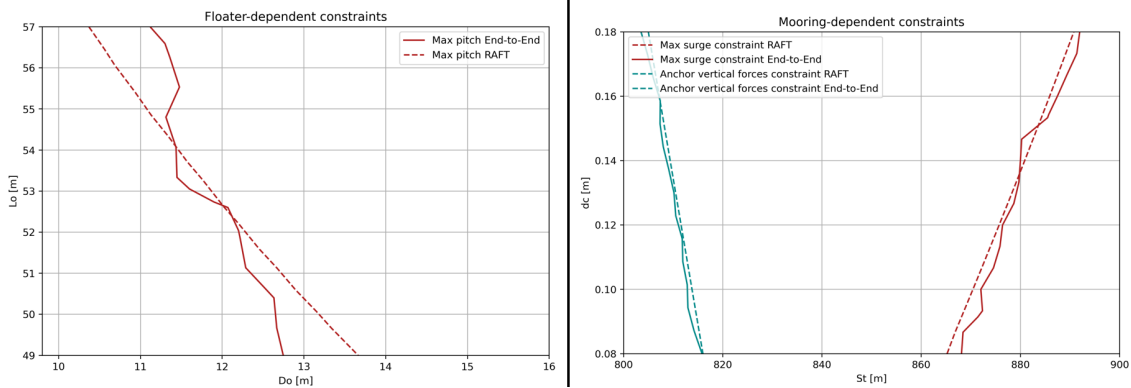
Figure 52: Constraint limits, calculated by RAFT and the "End-to-End" surrogate model, for the extreme environmental case.

5.3 Optimization results

With the two surrogate approaches examined, the optimization results are analyzed. Before demonstrating the optimization results of the two surrogate model-based workflows, it is important to investigate the optimization outcome produced by the same workflow when RAFT is utilized. For the following optimization results, a population of 30 individuals is optimized for 30 generations (iterations).

Figure 53 illustrates the Pareto front resulting from the optimization mentioned above. The first observation that can be made is that the trade-off of decreasing the standard deviation of power (improving power quality) is the increase in the substructure cost. Even though a line function can not accurately estimate the relationship between the two objectives, Table 11 shows the average cost increase for specific power deviation ranges. As shown, even though the optimum design utilized in a project depends on the requirements of the substructure, there are areas where the expense of reducing σ_P becomes less significant compared to others. For instance, the optimum designs in the [1050, 1100] KW range require a lower cost increase than those in the [930, 950] KW range.
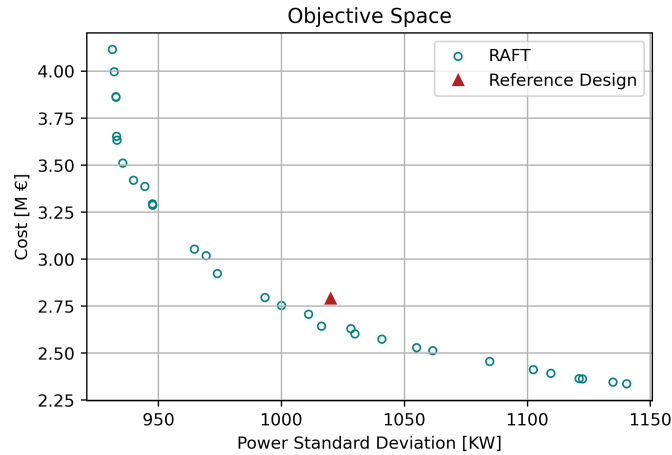
Figure 53: RAFT-based optimization Pareto Front, for a population of 30 individuals. Reference design indicated by red triangle.

Power deviation range [Kw]	935-950	950-1000	1000-1050	1050-1100	1100-1150
Average cost increase [Eur]	42942	10000	4047	2400	2000

Table 11: Average cost increase in specific power standard deviation ranges.

Regarding the Pareto front population, a 3D scatter plot combining every design variable of each individual can be found in the Appendix H. It becomes evident that, even though some characteristics of the Pareto population can already be identified, it is difficult to analyze the trends of the population thoroughly. In order to do so, the characteristics of the floater and the mooring lines are analyzed separately in the following sections. In parallel, the constraints that affect the final product of the optimization are also analyzed. Following the separate analysis, the way that the floater and mooring characteristics are combined is demonstrated.

5.3.1 Floater optimized population

Starting from the floater characteristics, Figure 54 illustrates the locations of the Pareto (optimized) points with respect to the design variables of the floater. In parallel, the objective functions' values and the maximum pitch constraint are included. Even though the contour plots are produced for a constant value of S_t and d_c, this illustration is representative of every different S_t and d_c combination. For the case of the power standard deviation, the colour bar is bounded to a

maximum of 1.1 MW. This is done in order to obtain a detailed plot of the area where the designs belonging to the Pareto front are located. As shown, The designs belonging to the Pareto front are clustered towards the top boundary of the design space. This boundary corresponds to floater designs of large pontoon lengths and a wide range of outer column diameters. The Pareto-front designs also cluster along the maximum pitch constraint boundary, indicating the importance of this constraint in determining the shape of the Pareto front.
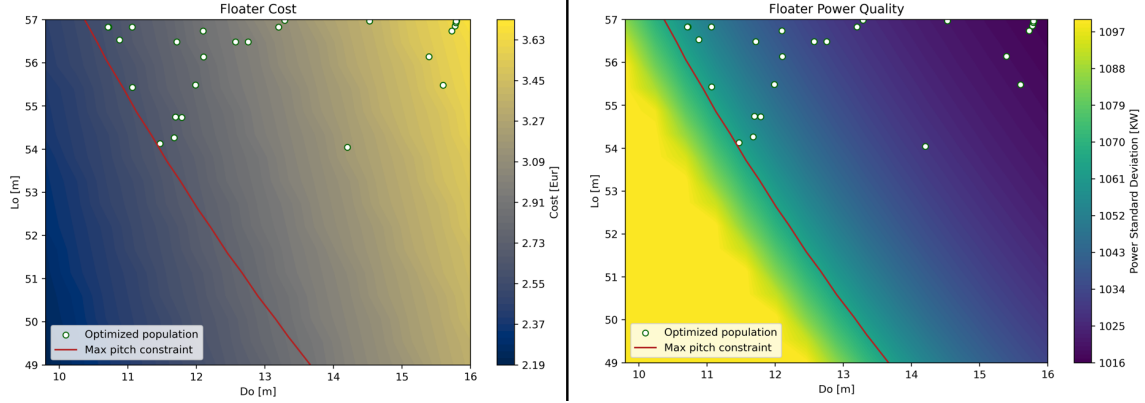
Figure 54: Pareto front population, with respect to the floater design variables. Mooring design variables are constant.

The reason for this distribution can be found in the cost and power quality contour plots. Following from the power quality contour plot, σ_P can be reduced (improving power quality) by either increasing the outer column diameter or increasing the inner-outer columns offset distance. The average decreasing rate of σ_P is 25.67 KW/m, when D_o increases, and 4.27KW/m, when L_o increases. Even though σ_P is affected approximately 6 times more by D_o than L_o, increasing the outer column diameter is dramatically more expensive than increasing the pontoons' length (which is the result of increasing L_o). Analytically, the average cost increase for a D_o and L_o increase is 177 and 31.5 K eur/m, respectively.

From examining the contour plots results, three main behaviours of the optimized population can be identified. These behaviours demonstrate how the optimizer exploits the characteristics of the design space. Firstly, the optimizer maximizes the inner-outer column offset distance, L_o, as this is the most economical way to decrease σ_P. Secondly, due to the significant effect of the outer column diameter on σ_P, the Pareto front designs are distributed along the entire range of Do. Increasing D_o leads to designs with further reduced σ_P, but with the downside of significant cost increase. Finally, for low D_o values, the maximum pitch angle becomes greater than 10°, violating the maximum pitch response constraint. Since D_o cannot be further reduced, several designs cluster along the maximum pitch constraint boundary, achieving reduced cost (by reducing L_o), with a moderate increase in σ_P.

5.3.2 Mooring lines optimized population

Similarly to the analysis of the floater design variables (Section 5.3.1), Figure 55 illustrates the locations of the Pareto front points with respect to the design variables of the floater. For the mooring characteristics, the trend followed by the optimized population is towards the bottom boundary of the design space and the maximum surge constraint limit.

Based on the cost contour, it is evident that modifications in the total length of the mooring line have insignificant effects on the cost of the mooring lines compared to modifying the chains' thickness. To demonstrate, the average cost increase of the mooring lines is 5120 eur/mm, for an increase in d_c, and 385 eur/m, when increasing S_t. Regarding power quality, there is not a single trend between the mooring design variables and σ_P. Specifically, for a given value of d_c, σ_P is

maximized for S_t values in the [820, 840] range and gradually reduces around this area. Another observation is that, among the floater's and mooring lines' design variables, the floater's variables have a more decisive effect on σ_P. Even more specifically, D_o is the floater's parameter affecting power deviation the most. Unlike power deviation, cost is affected by both the floater and mooring configurations. Even though the effects of configurations on the floater affect cost more than the mooring ones, both are considered significant. Finally, it is worth mentioning that the power standard deviation contour plot differs for different values of the floater's design variables. If the contour plot of σ_P is observed, there is a noticeable succession of contour peaks. For different floater designs, the location of the peak shifts along the x-axis. Nevertheless, the way the contour plot is shaped around those peaks follows the same trends.

From the assessment of the contour plots results, three main behaviour of the optimizer can be identified. Firstly, the optimizer favours designs that minimize the thickness of the mooring lines. That way, the most economical design is chosen for a certain σ_P. However, for S_t values higher than approximately 865 m, in combination with the minimum chain diameter, the maximum surge constraint becomes active. In this area, the mooring chains' diameter is increased, as increasing d_c leads to a reduction of σ_P, although at the cost of a significant increase in cost. Finally, due to the maximum vertical tension constraint at the touch-down point for mooring line lengths of less than approximately 815 m, no designs with mooring lengths lower than 815 m are utilized by the floater.
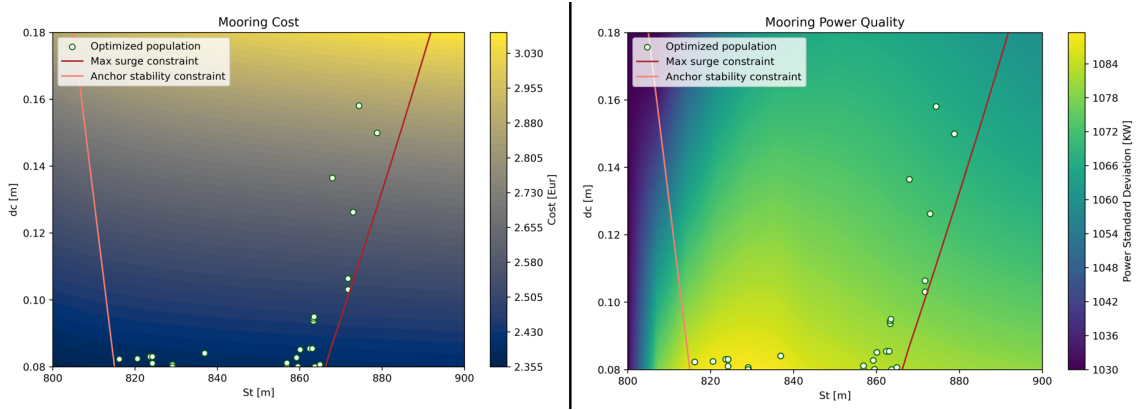
Figure 55: Pareto (optimized) population, with respect to the mooring design variables. Floater design variables are constant.

5.3.3 Floater and mooring combination

After conducting a separate analysis of the floater and mooring variables, the driving factors behind the optimized designs utilized by the optimizer have been addressed separately. What is missing from the Pareto front analysis is understanding which designs correspond to which Pareto points. To investigate the correspondence of the optimum designs, the Pareto front points are divided into 5 different coloured zones, as shown in Figure 56.

Regarding the floater design variables, as σ_P decreases, there is a clear trend of the optimized population towards increased D_o values. Based on the contour plots of Figure 54, this D_o increase results in the reduction observed in σ_P, but with the drawback of significantly affecting the cost of the floater. Furthermore, even though the optimizer tends to keep the value of L_o close to the upper boundary, there are cases of design configurations with lower L_o values. These designs can be observed in the orange and green zones. According to the morphology of the contour plots, a reduction in L_o combined with a slight D_o increase leads to a significant reduction of σ_P. This combination also leads to a moderate increase in cost while still fulfilling the maximum pitch constraint. However, when such designs are observed, floater configurations of similar D_o and higher L_o follow, minimizing power deviation. For the designs corresponding to the purple zone, the floater designs of this zone correspond to the same floater (practically). This outcome is based on the fact

that the optimizer has reached the maximum boundaries of both D_o and L_o. Therefore, in the purple zone, the Pareto front designs are determined by variations of the moorings design variables.

For the mooring lines characteristics, the higher σ_P zone (orange coloured) is located in the area defined within the [828, 858] S_t range and close to the lower boundary (minimum d_c). This region corresponds to the highest σ_P (see Figure 55). Inside this region, the value of d_c is minimized, while S_t is spread, as it results in minor effects on cost. With σ_P decreasing (green and yellow zones), the design points spread horizontally from the orange zone. Based on Figure 55, it becomes clear that this movement is based on the fact that, close to the orange zone, the σ_P contours decrease in every direction, improving power quality (Figure 55). According to the cost contours, S_t modifications are more economical compared to d_c ones. Nevertheless, cases where d_c is increased, which reduces σ_P, are also identified. However, these d_c modifications are consistently minor due to the high-cost drawback. Interestingly, there are mooring designs identified in the green and yellow zone with lower costs and decreased σ_P compared to the orange zone. The reason why this reduction in cost is not demonstrated on the Pareto front is that, for insignificant changes in d_c, the total substructure cost is dominated by the floater design, and specifically by D_o. Even more generally, among the 4 design variables, S_t is the one affecting cost the least.

In the blue zone, a new trend takes place. As the optimizer exploits the whole range of mooring designs that keep d_c close to the boundary while S_t spreads further away from the orange zone, the maximum surge and maximum vertical tension on the anchor constraints are met. Due to the morphology of the power quality contours (Figure 55), design configurations of lower cost and σ_P cluster around this area. Even though increasing d_c significantly affects the cost of the mooring lines (and the substructure), σ_P continues to decrease. Finally, in the purple zone, the cost of the substructure increases drastically, following the increasing rate of d_c. Unlike cost, though, this d_c increase leads to minor reductions on σ_P. Since the purple zone floater designs have reached the upper right boundary of the design space, σ_P is not affected any more by the floater design. Until this point, the floater design variations are the main driver for σ_P variation. As a matter of fact, the σ_P variations observed in the purple zone are minor.
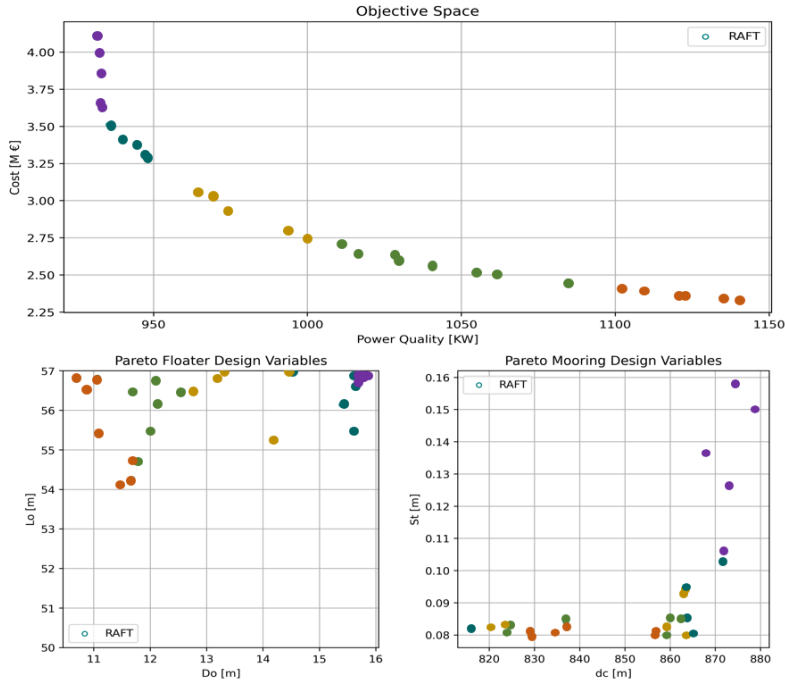
Figure 56: Division of the optimum population into different zones.

5.4 "PyHAMS" and "End-to-End"-based optimization

With the optimization results analyzed in Section 5.3, the same optimization problem is solved by utilizing the "PyHAMS" and "End-to-End"-based algorithms. In the following paragraphs, the main differences between the three results and the potential sources of errors are addressed.

Figure 57 illustrates the Pareto fronts produced by the three alternative optimization workflows. Based on the three Pareto fronts, it becomes clear that the "PyHAMS"-based optimization is able to reproduce the RAFT-based optimization result accurately, as their Pareto fronts are almost identical. This is an expected outcome, since the error between the optimization-related outputs of those two models is minor, as shown in Figure 45. Regarding the "End-to-End"-based optimization, the Pareto front computed by the optimizer shows a similar trend to the RAFT-based one. However, it is also evident that the "End-to-End"-based Pareto front underestimates the cost of the substructure, for a given power standard deviation, shifting the Pareto down. This is related to the computational errors produced by the "End-to-End" approach (Figure 48 and Figure 49).

In order to further investigate the factors that lead to the Pareto fronts demonstrated, the designs populating the fronts obtained with the three different approaches are examined.
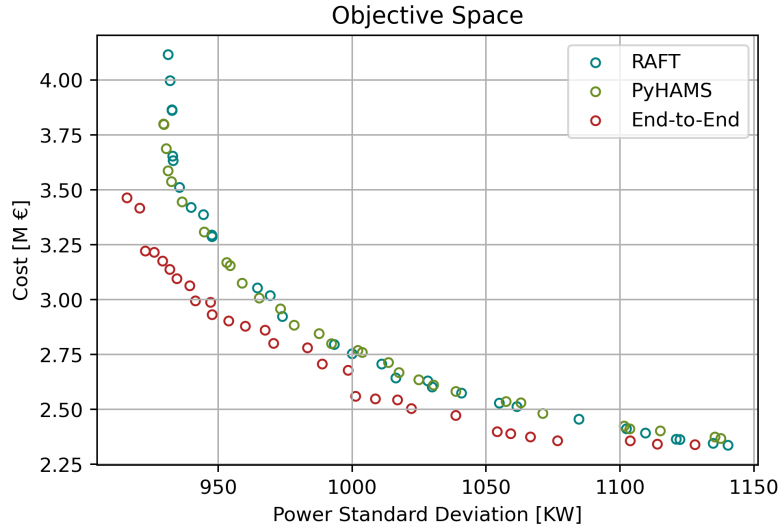
Figure 57: Pareto fronts produced by the same optimization problem by utilizing the three different algorithms.

5.4.1 Optimized populations floater

The comparison between the designs populating the Pareto front obtained with the "PyHAMS" and the "End-to-End" approach is carried out first with respect to the floater's design variables and then with respect to the moorings design variables. As the "PyHAMS" front is very similar to the RAFT-derived front, the main focus of this section is to understand the factors that differentiate the "End-to-End" front. However, the "PyHAMS"-derived optimum population is also included. The main reason for this choice is to demonstrate that even though the Pareto fronts are almost identical, the optimized population designs are not the same. This is a natural outcome, as the NSGA-II algorithm driving the optimization process is a stochastic algorithm. For that reason, every time the same optimization problem is repeated, the resulting Pareto front is not identical to the previous ones. The differences observed, however, are minor. As a secondary reason, even though the "PyHAMS" approach's errors are insignificant, the designs evaluated and selected within every iteration can be affected. Thus, instead of converging on the same design configurations, the two workflows converge to designs that are within the same region. Finally, the mismatches between the RAFT and "PyHAMS"-based populations can be used as a reference to Pareto population differences that can be considered insignificant.

Starting from the floater design variables, Figure 58 illustrates the optimum populations of the three optimization workflows with respect to the floater's characteristics. As shown, the optimized population produced by the "PyHAMS"-based optimization follows the same trends as the RAFT-based one. Unlike the "PyHAMS"-derived population, the optimized population computed by the "End-to-End"-based optimization shows some noticeable deviation from the optimized population. The "End-to-End" population shows a similar tendency towards the upper bound of L_o, while exploiting a wide range of different D_o values. However, the L_o values utilized are lower (on average) than the ones selected by the other two optimization workflows. Additionally, even though the optimization algorithm exploits a wide range of D_o values, there are no designs with outer column diameters of less than 11.5 m.

Figure 58: Optimized populations, produced by the three different optimization workflows, with respect to the floater design variables.

For the "End-to-End"-based optimization of the floater design variables, there are two key factors that can affect the optimization result: the error in calculating the maximum pitch response and the power standard deviation. For the maximum pitch response, Figure 59b demonstrates the maximum pitch response constraint limit with respect to the floater optimum designs. The constraint shape is almost identical to the one observed in Figure 52 (minor differences are the result of different mooring design configurations), verifying that the maximum pitch constraint limit is defined by the extreme environmental case. For $L_o \geq 54$, the maximum pitch constraint is overestimated, whereas, for $L_o \leq 54$, the constraint is underestimated. Focusing on the optimized population, the overestimation of $\chi_{5,max}$ for $L_o \geq 54$ prevents the optimizer from exploiting designs with D_o less than approximately 11.5 m.

Even though the Pareto front designs follow the same trend of maintaining L_o values close to the top boundary, the L_o values used are (on average) lower compared to the ones shown in the RAFT and "PyHAMS"-based Pareto fronts. By considering the power deviation contour plot (Figure 59a), it is observed that the standard deviation of power is not significantly affected by L_o. Even though an increase in L_o still results in a reduction of σ_P, the reduction rate is not gradual, as shown in Figure 72. Considering that, the Pareto population is clustered in a region of greater L_o range, due to the fact that the σ_P reduction offered by increasing L_o close to the boundary is not as important as in the RAFT-based optimizations.
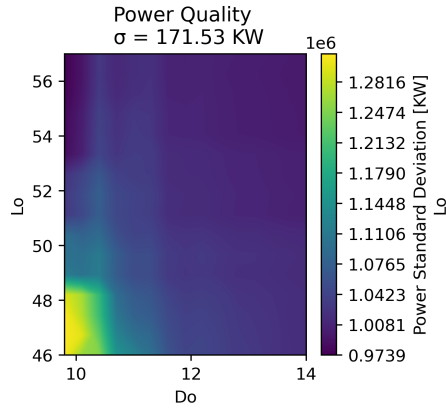
(a) Power deviation contour plot, with respect to the floater design variables, produced by the "End-to-End" approach.
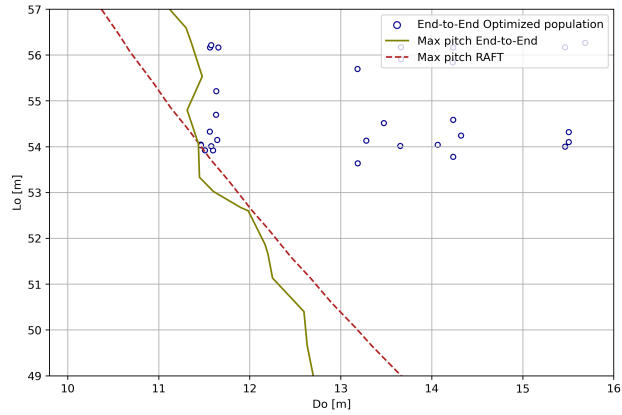
(b) Maximum pitch constraint computed by the "End-to-End" approach.

Figure 59: Pareto population driving factors, for the "End-to-End"-based optimization.

5.4.2 Optimized populations mooring lines

Respectively, for the mooring lines' design variables, Figure 55 demonstrates the designs populating the Pareto front for the cases where the design analysis is carried out with the "End-to-End" surrogate model, the "PyHAMS" surrogate model, and RAFT. In all the cases analysed, the Pareto front designs tend to cluster towards the lower boundary of the d_c range, with d_c reduction being a significant driver for cost reduction.
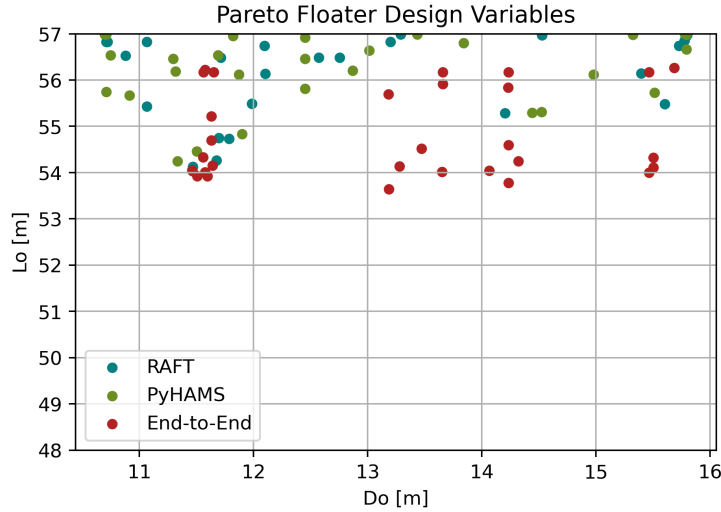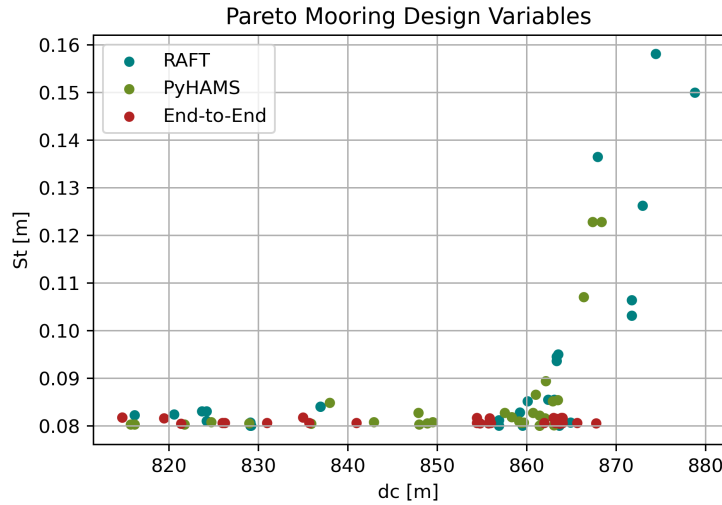

Figure 60: Optimized populations, produced by the three different optimization workflows, with respect to the mooring lines' design variables.

However, for lengths higher than approximately 860 meters, RAFT and the "PyHAMS"-based optimum designs cluster along the portion of the design space where the maximum surge constraint is active. For the "End-to-End" surrogate, the designs populating the Pareto front cluster along the lower boundary of the dc range even for St higher than 860. Figure 61 shows the maximum surge constraint limit, as predicted by RAFT and by the "End-to-End" surrogate model. The same Figure also shows the designs populating the Pareto front obtained with the "End-to-End" approach. Based on it, the maximum surge constraint boundary is slightly shifted to the right, due to the "End-to-End" approach underestimating the maximum surge displacement. To a certain extent, this shift can affect the location of the optimized mooring configurations, as the boundary's interaction with the optimized population is lower compared to RAFT. Nevertheless, the most

important factor affecting the optimization result is the NSGA-II algorithm, which is inherently stochastic.

With the Pareto front population of the "End-to-End"-based algorithm analyzed, the factors that drive the "End-to-End" Pareto front below the RAFT-derive front are clear. The Pareto population calculated by the "End-to-End"-based algorithm combined lower pontoon lengths (on average), together with very low mooring lines thickness. This combination leads to a significant underestimation of the substructure cost, for a given σ_P value.
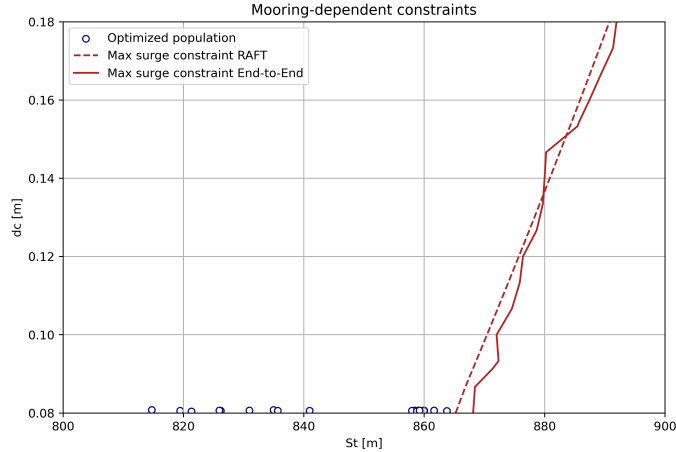
Figure 61: Maximum surge constraint computed by the "End-to-End" approach.

5.5 Computational resources

Implementing a surrogate model in the optimization workflow aims to diminish the large computational resources required to run a RAFT-based optimization. To examine the computational resources needed for each optimization process, the same optimization problem (30 individuals, 30 iterations) is solved under the same conditions in order to monitor the optimization time needed. Thus, the computational resources required are evaluated by monitoring the computational time. The less the computational time, the less the computational resources required.

The optimization process is conducted in a computer equipped with a 6-core (12 threads) 5.0 GHz CPU, a RAM of 32 GB and a GPU of 4 GB. For optimization processes that require more time, the temperature of the system is also monitored. That ensures that the computer does not slow down due to thermal throttling.

Table 12 compares the computational time needed for the optimization when the substructure analysis is based on RAFT, the "PyHAMS" surrogate model, and the "End-to-End" surrogate model. When considering only the optimization time, the computational reduction achieved by both surrogate models is dramatic, reducing the optimization time required from days to hours and even minutes. This efficiency increase makes conducting the optimization with conventional processing units (i.e. a laptop) feasible.

	RAFT	"PyHAMS"	"End-to-End"
Computational time	2.8 days	6.5 hrs	12 mins
Computational time ratio compared to RAFT	-	10.3%	0.03%

Table 12: Computational time comparison between the three alternative optimization approaches.

Nevertheless, the above-mentioned computational times do not account for the time needed to construct the training datasets. Table 13 demonstrates the time required by the same optimization

problem when the computational time of producing the training datasets is taken into account. Additionally, Figure 62 shows the time needed from each specific process (optimization and training dataset). When the dataset production time is included, the "End-to-End" approach requires 1.95 times more computational time compared to the RAFT-based optimization for a single optimization run. Moreover, if the accuracy of the model needs to be improved, then the size of the dataset needs to be expanded. That increases computational time even further. However, in the case of optimizing more than one substructure, the "End-to-End" approach can potentially reduce the computational time needed.

	RAFT	"PyHAMS"	"End-to-End"
Computational time	2.8 days	1.6 days	5.46 days
Computational time ratio compared to RAFT	-	43 %	195 %

Table 13: Computational time comparison between the three alternative optimization approaches, by also including the training dataset production time
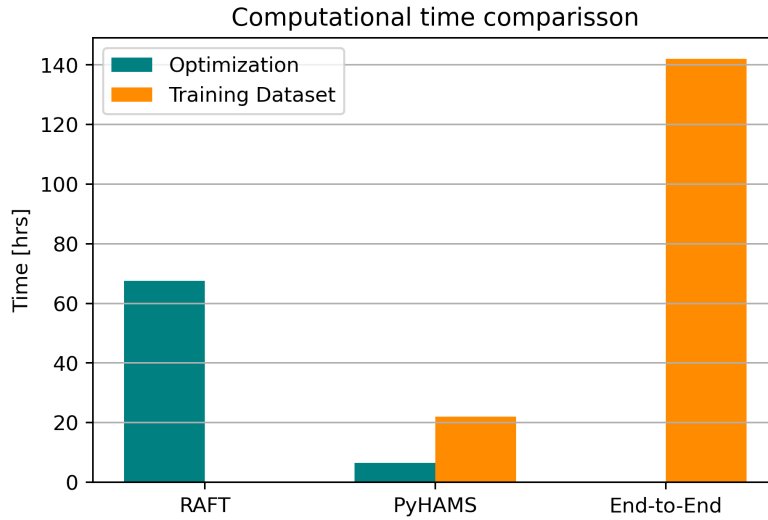
Figure 62: Computational time comparison between the three optimization workflows.

Even by including the computational time of producing the training dataset, the "PyHAMS" approach is still more efficient than the RAFT-based optimization, even for a single optimization run. Thus, the "PyHAMS" surrogate model is the most efficient approach, achieving a 43.6% computational time reduction for this specific optimization problem. If an optimization of fewer individuals is investigated, specifically an optimization requiring less than 350 substructure simulations in total (which is the size of the training dataset), the RAFT-based optimization is the most efficient choice.

5.6 "PyHAMS" and "End-to-End" approaches comparison

With the computational resources needed by the two surrogate-based optimization workflows analyzed, it becomes evident that each one of the two surrogate-based approaches holds some strong advantages compared to the other. Even though the most suitable optimization workflow depends on the problem-specific optimization requirements, this section lists the most important factors that differentiate the two approaches.

Considering the structure of the two approaches and the above-mentioned results:

- Regarding the accuracy of the two approaches, the "PyHAMS" surrogate model prediction accuracy is higher than the "End-to-End" surrogate. This is proven by both the deviation error plots (Section 5.1.3, Section 5.2.3) and the optimization results' comparison (Section 5.4). As a matter of fact, the "PyHAMS" approach predictions are very reliable and match very well the results obtained by using the full RAFT workflow.

- The effectiveness of the two approaches in terms of computational time depends on the specific characteristics of the optimization. If the training dataset already exists, to "End-to-End"-based algorithm is radically more efficient than the "PyHAMS"-based one, calculating the optimization result within minutes. However, since the training dataset needs to be produced in most cases, the "End-to-End"-based optimization workflow is difficult to be justified for the optimization of a single substructure design. The reason for that is the extensive computational resources needed to produce the training dataset. On the contrary, the "PyHAMS"-based optimization is very suitable for the optimization of a single design (or even less than 5) due to the fact that the computational resources needed for producing the dataset are less than conducting the optimization on a RAFT-based algorithm. It is worth noting, however, that this statement is based on the assumption that the optimization problem will require a large number of simulations. If the optimization problem's desired output is obtaining a Pareto front that consists of a small number of optimum designs, and it is expected that the simulations needed are less than 350, then the most efficient way to solve the problem is by utilizing the RAFT-based optimization.

- The "PyHAMS"-based workflow is more versatile. The workflow is more versatile due to the fact that there are fewer inputs used in the "PyHAMS"-based algorithm. Unlike the "End-to-End" algorithm, which contains seven inputs, the "PyHAMS" training dataset contains only two inputs. This implies that the limitations of the "PyHAMS" model are floater designs that require D_o and L_o values outside of the training dataset's boundaries. The second limitation of "PyHAMS" is that additional design parameters of the floater, such as the draft, cannot be used as design variables. More importantly, the additional limitations of the "End-to-End" approach can potentially affect the model's area of application. Firstly, due to the limitations of the environmental condition inputs, there are environmental cases that the "End-to-End" approach is not able to make a prediction with the expected accuracy. Also, the mooring design parameters limitations can affect the "End-to-End" approach's capabilities when the water depth changes. For instance, for a water depth of 350 m., it is expected that the upper limit of $L_o \leq 900m$ can affect the maximum mooring length that the optimization workflow can exploit. This is particularly important for the case of the optimization of every substructure inside a wind farm. In such a case, the water depth variations can be quite significant. On the contrary, as long as the substructure optimized is designed to support the IEA 15 MW wind turbine, the limits on the floater design variables can be considered very conservative.

- From a more practical point of view, the "PyHAMS" algorithm is expandable to a certain degree. Unlike the "End-to-End" surrogate model, the surrogate model implemented in the "PyHAMS" approach replaces a specific analysis block of RAFT. For that reason, the remaining analysis blocks inside RAFT can be modified according to the user's needs. For instance, a promising design parameter that can further improve the optimization results is the radial distance of the anchor (X_{tot}). Due to the structure of the "PyHAMS" approach's algorithm, modelling the mooring lines is analyzed by MoorPy and does not affect the surrogate model replacing "PyHAMS". In fact, even a radical configuration in the mooring system,

such as substituting it with a taut mooring system, is feasible. Thus, the mooring-related design parameters that are optimized can be expanded without the need for a new surrogate model. Similarly, the optimization problem's constraints are not directly derived from the surrogate model. This is why, if a specific constraint needs to be modified or a new constraint is needed, it can be introduced to the problem without the need for a new surrogate model. The only limitation to the expansion capabilities of the "PyHAMS" approach comes from design parameters that affect the hydrodynamic properties of the floater. In other words, if a new floater-related design parameter needs to be introduced to the optimization, then the existing surrogate model needs to be modified accordingly.

The comparison of the two approaches is summarized in Table 14.

	"End-to-End"	"PyHAMS"
Error	Noticable	Insignificant
Computational efficiency	$N_{sim} \geq 3000$	$N_{sim} \geq 1000$
Versatility	Limited	Fully (practically)
Expandable	No	Yes

Table 14: "PyHAMS" and "End-to-End" approaches comparison.

6 Conclusion and Recommendations

6.1 Conclusion

This thesis research delves into exploring the trade-off between power quality and cost in optimizing a semisubmersible floating wind turbine platform. In parallel, it investigates the viability of substituting the computationally demanding frequency domain analysis block of the optimization process with a computationally lighter surrogate model. The objectives are to establish the Pareto front of cost and power quality and enhance the optimization efficiency by implementing the surrogate model. To accomplish this, an optimization workflow for the semisubmersible floater is created based on an altered version of the VolturnUS-S reference floater. The design parameters under investigation are the outer column diameters, the offset distance between the inner and outer column, the length of the mooring lines, and the chain thickness of the mooring lines. Lastly, two distinct surrogate model approaches are implemented into the optimization workflow. The first approach substitutes the entire frequency domain model (NREL's RAFT) with a surrogate model, while the second one replaces one of RAFT's analysis blocks with a surrogate model.

The results of this study indicate that the trade-off of enhancing the power quality of a floating wind turbine is the cost increase of the substructure. Adjustments to the offset distance of the columns and length of the mooring lines are the most cost-effective methods to improve power quality. Modifying the diameter of the outer columns has a significant impact on power quality but also raises the substructure cost. regarding the thickness of the mooring lines, it is maintained at low values. This is based on the fact that power quality can be improved similarly by configurations either to the mooring lines' thickness or the total mooring length. Among the two options, configurations on the total mooring length are more cost-efficient.

Among the two surrogate models, the first one aims to replace the frequency domain model entirely. This "End-to-End" approach's significantly enhances computational efficiency to the extent that the optimization process can be accomplished within a matter of minutes. Nonetheless, a trade-off exists as the surrogate model-based algorithm compromises the accuracy of predicting the frequency domain analysis outputs, as opposed to analytically computing them. Such inaccuracies lead to errors in the optimization process, thereby affecting the final outcome. Additionally, even though the optimization algorithm's efficiency has increased, the surrogate model requires a large training dataset, which needs significantly more computational resources compared to those needed for the optimization of a single design.

The second surrogate model substitutes PyHAMS, a specific analysis block inside RAFT that requires most of the computational resources of the optimization process. Using a surrogate model instead of PyHAMS shows a significant decrease in computational time. Although this method does not result in the same reduction as the End-to-End model, the error introduced is insignificant. Consequently, the optimization workflow using the PyHAMS approach generates nearly identical results as the RAFT-based optimization algorithm. Additionally, this model requires a smaller dataset compared to the End-to-End model. Therefore, the PyHAMS surrogate model approach is the only method that, even by accounting for the training dataset production time, requires less computational resources than the RAFT-based algorithm. Nevertheless, the efficiency that can be achieved by each model depends on the specific characteristics of the optimization problem. Depending on the number of simulations needed by the optimization process, the computational efficiency of the two approaches can differ.

6.2 Future work

In this thesis, the optimization workflow and surrogate models have been developed based on certain assumptions and are subject to limitations. However, there is potential for future work that could greatly enhance the results of the thesis. The following paragraphs highlight some of these potential areas for further exploration.

Starting from the objectives of the optimization, it is clear that the optimization objectives of this thesis do not prioritize the substructure's structural integrity. This is the reason why the optimized population is represented by design configurations located at the lower boundary of the mooring lines thickness and the upper boundary of the columns offset distance (which leads to long pontoons). If the optimization workflow aims to minimize substructure fatigue, these designs are expected to accumulate significant fatigue, which impacts the optimization result. Alternatively, the structural integrity of the substructure could be regulated by constraints. For instance, a constraint ensuring that the accumulated fatigue on the mooring lines does not exceed a specific limit would block the optimizer from exploiting design configurations of very low chain thicknesses. Finally, in order to estimate cost and structural integrity with a higher level of fidelity, deriving specific dimensions of the substructure (for instance, the pontoon wall thickness) based on the dynamic loads applied on the substructure is a promising addition. In order to implement such an addition, an initial simulation is needed, in order to compute dynamic loads, based on which the substructure dimensions can be calculated. In such a case, the need for an efficient way to conduct simulations is even greater.

Increasing the number of design variables is another way to obtain a more well-optimized design. Every time a new design variable is introduced, new design configurations can be assessed by the optimizer to reduce the objective functions. Some promising design variables that could have a significant effect on cost and power quality are the radial distance of the anchorage point, the draft of the floater, or the radius of the cylindrical pontoons. However, in order to avoid the exploration of unrealistic designs, additional constraints should also be introduced to the optimization problem. Especially for the case of the PyHAMS approach, increasing the number of mooring-related design variables does not require setting up a new surrogate model, making the implementation easier from a practical point of view.

In the case of the End-to-End approach, 5 different surrogate models have been examined. Even though among these models, the XGBoost surrogate model achieves the highest accuracy, there are numerous models that have not been tested in this research. Furthermore, even among the current models, some of those, like artificial neural networks, are heavily affected by user-defined parameters. Thus, there is potential in either improving the current models or examining new ones. This future work suggestion is particularly important for the End-to-End approach since the current model's error can not be considered minor.

In this thesis, the NSGA-II algorithm, which is a GA-based stochastic algorithm, was used in the optimization. While the algorithm effectively converges towards the Pareto front by minimizing costs and enhancing power quality, there are numerous alternative optimization algorithms, some of which claim to be more efficient than the NSGA-II. Additionally, the current algorithm eliminates design configurations that violate constraints, which, while offering a stable convergence, can be slow. To increase efficiency, an alternative approach could be to define a penalization function for the violation designs, which can lead to faster convergence towards the constraints with proper tuning of penalization parameters. The advantage of this recommendation is that modifying the optimization algorithm does not affect other components of the workflow (design variables, constraints, objective functions etc.).

Bibliography

[1] *European Comision , Renewables Energy Target*. URL: https://energy.ec.europa.eu/topics/renewable-energy/renewable-energy-directive-targets-and-rules/renewable-energy-targets_en#the-2030-targets (visited on 27th July 2023).

[2] J Hamilton. *Lazard's levelized cost of energy analysis-version 15.0*. 2021.

[3] *Ørsted: Can Wind Turbines Be Recycled?* URL: https://orsted.com/en/insights/the-fact-file/can-wind-turbines-be-recycled (visited on 27th July 2023).

[4] *LM Wind: Zero wastE Blade ReseArch (ZEBRA) Project*. URL: https://www.lmwindpower.com/en/sustainability/blade-recycling (visited on 27th July 2023).

[5] *Wind Europe*. URL: https://windeurope.org/ (visited on 12th July 2023).

[6] Michael Borg et al. 'Technical definition of the tetraspar demonstrator floating wind turbine foundation'. In: *Energies* 13.18 (2020), p. 4911.

[7] Alberto Ghigo et al. 'Platform optimization and cost analysis in a floating offshore wind farm'. In: *Journal of Marine Science and Engineering* 8.11 (2020), p. 835.

[8] Matilda Kreider et al. 'Offshore Wind Energy: Technology Below the Water [Slides]'. In: (July 2022). DOI: 10.2172/1877072. URL: https://www.osti.gov/biblio/1877072.

[9] Odd Faltinsen. *Sea loads on ships and offshore structures*. Vol. 1. Cambridge university press, 1993.

[10] Mareike Leimeister and Philipp Thomas. 'The OneWind® Modelica library for floating offshore wind turbine simulations with flexible structures'. In: *Proceedings of the 12th International Modelica Conference, Prague, Czech Republic, May 15-17, 2017*. 132. Linköping University Electronic Press. 2017, pp. 633–642.

[11] Dong-Min Park et al. 'Study on added resistance of a tanker in head waves at different drafts'. In: *Ocean Engineering* 111 (2016), pp. 569–581.

[12] Martin OL Hansen. *Aerodynamics of wind turbines*. Routledge, 2015.

[13] GLAS DNV. 'Offshore Mooring Chain-DNVGL-OS-E302'. In: *Offshore Stand* (2015), pp. 22–23.

[14] Nikhar J Abbas et al. 'A reference open-source controller for fixed and floating offshore wind turbines'. In: *Wind Energy Science* 7.1 (2022), pp. 53–73.

[15] Todd R Kelley. 'Optimization, an important stage of engineering design'. In: *The Technology Teacher* 69.5 (2010), p. 18.

[16] Joaquim RRA Martins and Andrew Ning. *Engineering design optimization*. Cambridge University Press, 2021.

[17] J. Blank and K. Deb. 'pymoo: Multi-Objective Optimization in Python'. In: *IEEE Access* 8 (2020), pp. 89497–89509.

[18] Manoj Kumar et al. 'Genetic algorithm: Review and application'. In: *Available at SSRN 3529843* (2010).

[19] S Guo. *An introduction to surrogate optimization: Intuition, illustration, case study, and the code*. 2020.

[20] Ping Jiang, Qi Zhou and Xinyu Shao. *Surrogate model-based engineering design and optimization*. Springer, 2020.

[21] Douglas C Montgomery, Elizabeth A Peck and G Geoffrey Vining. *Introduction to linear regression analysis*. John Wiley & Sons, 2021.

[22] Yan-Yan Song and LU Ying. 'Decision tree methods: applications for classification and prediction'. In: *Shanghai archives of psychiatry* 27.2 (2015), p. 130.

[23] Claude Elwood Shannon. 'A mathematical theory of communication'. In: *The Bell system technical journal* 27.3 (1948), pp. 379–423.

[24] Leo Breiman. 'Random forests'. In: *Machine learning* 45 (2001), pp. 5–32.

[25] Gérard Biau and Erwan Scornet. 'A random forest guided tour'. In: *Test* 25 (2016), pp. 197–227.

[26] Thomas G Dietterich. 'Ensemble methods in machine learning'. In: *Multiple Classifier Systems: First International Workshop, MCS 2000 Cagliari, Italy, June 21–23, 2000 Proceedings 1*. Springer. 2000, pp. 1–15.

[27] Tianqi Chen and Carlos Guestrin. 'Xgboost: A scalable tree boosting system'. In: *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining.* 2016, pp. 785–794.

[28] Christopher M Bishop and Nasser M Nasrabadi. *Pattern recognition and machine learning.* Vol. 4. 4. Springer, 2006.

[29] Mareike Leimeister et al. 'Larger MW-class floater designs without upscaling?: A direct optimization approach'. In: *International Conference on Offshore Mechanics and Arctic Engineering.* Vol. 58769. American Society of Mechanical Engineers. 2019, V001T01A032.

[30] John Marius Hegseth, Erin E Bachynski and Joaquim RRA Martins. 'Integrated design optimization of spar floating wind turbines'. In: *Marine Structures* 72 (2020), p. 102771.

[31] Nicolò Pollini et al. 'Gradient-based optimization of a 15 MW wind turbine spar floater'. In: *Journal of Physics: Conference Series.* Vol. 2018. 1. IOP Publishing. 2021, p. 012032.

[32] Serag-Eldin Abdelmoteleb et al. 'Preliminary Sizing and Optimization of Semisubmersible Substructures for Future Generation Offshore Wind Turbines'. In: *Journal of Physics: Conference Series.* Vol. 2362. 1. IOP Publishing. 2022, p. 012001.

[33] Christopher Allen et al. *Definition of the UMaine VolturnUS-S reference platform developed for the IEA Wind 15-megawatt offshore reference wind turbine.* Tech. rep. National Renewable Energy Lab.(NREL), Golden, CO (United States); Univ. of . . ., 2020.

[34] Giulio Ferri et al. 'Platform and mooring system optimization of a 10 MW semisubmersible offshore wind turbine'. In: *Renewable Energy* 182 (2022), pp. 1152–1170.

[35] Amy Robertson et al. *Definition of the semisubmersible floating system for phase II of OC4.* Tech. rep. National Renewable Energy Lab.(NREL), Golden, CO (United States), 2014.

[36] Jae-chul Lee et al. 'An optimal sub-structure for a TLP-type wind turbine based on neuro-response surface method'. In: *Journal of Marine Science and Technology* 20.4 (2015), pp. 604–616.

[37] Jason Jonkman et al. *Definition of a 5-MW reference wind turbine for offshore system development.* Tech. rep. National Renewable Energy Lab.(NREL), Golden, CO (United States), 2009.

[38] Matthew Hall, Brad Buckham and Curran Crawford. 'Evolving offshore wind: A genetic algorithm-based support structure optimization framework for floating wind turbines'. In: *2013 MTS/IEEE OCEANS-Bergen.* IEEE. 2013, pp. 1–10.

[39] Mareike Leimeister, Maurizio Collu and Athanasios Kolios. 'A fully integrated optimization framework for designing a complex geometry offshore wind turbine spar-type floating support structure'. In: *Wind Energy Science Discussions* (2020), pp. 1–35.

[40] Benoit de Renty. 'Gradient-based design optimization of a semi-submersible floating wind turbine'. MA thesis. NTNU, 2020.

[41] Jeong Du Kim and Beom-Seon Jang. 'Application of multi-objective optimization for TLP considering hull-form and tendon system'. In: *Ocean Engineering* 116 (2016), pp. 142–156.

[42] Suguang Dou et al. 'Optimization of floating wind turbine support structures using frequency-domain analysis and analytical gradients'. In: *Journal of Physics: Conference Series.* Vol. 1618. 4. IOP Publishing. 2020, p. 042028.

[43] E Faraggiana et al. 'An efficient optimisation tool for floating offshore wind support structures'. In: *Energy Reports* 8 (2022), pp. 9104–9118.

[44] Meysam Karimi et al. 'A multi-objective design optimization approach for floating offshore wind turbine support structures'. In: *Journal of Ocean Engineering and Marine Energy* 3.1 (2017), pp. 69–87.

[45] M Collu and M Borg. 'Design of floating offshore wind turbines'. In: *Offshore wind farms*. Elsevier, 2016, pp. 359–385.

[46] Torgeir Moan et al. 'Recent Advances in Response Analysis of Floating Wind Turbines in a Reliability Perspective'. In: *International Offshore Wind Technical Conference (IOWTC)*. 2018.

[47] Barry Johnston et al. 'Levelised cost of energy, A challenge for offshore wind'. In: *Renewable Energy* 160 (2020), pp. 876–885.

[48] Antonio Pegalajar-Jurado, Michael Borg and Henrik Bredmose. 'An efficient frequency-domain model for quick load analysis of floating offshore wind turbines'. In: *Wind Energy Science* 3.2 (2018), pp. 693–712.

[49] Matthew Hall et al. 'An Open-Source Frequency-Domain Model for Floating Wind Turbine Design Optimization'. In: *Journal of Physics: Conference Series*. Vol. 2265. 4. IOP Publishing. 2022, p. 042020.

[50] Adebayo Ojo, Maurizio Collu and Andrea Coraddu. 'Multidisciplinary design analysis and optimization of floating offshore wind turbine substructures: A review'. In: *Ocean Engineering* 266 (2022), p. 112727.

[51] Ajit C Pillai, Philipp R Thies and Lars Johanning. 'Mooring system design optimization using a surrogate assisted multi-objective genetic algorithm'. In: *Engineering Optimization* (2018).

[52] Milad Hassani. 'Hydrodynamic performance optimization of semi-submersible floaters for offshore wind turbines'. MA thesis. uis, 2022.

[53] Martin Mathiesen, Anja K Meyer and B Kvingendal. *Hywind Buchan Deep Metocean Design Basis RE2014-002*. Tech. rep. Tech. rep., Norway, 2014.

[54] Peter Hauge Madsen and DTU Risø. 'Introduction to the IEC 61400-1 standard'. In: *Risø National Laboratory, Technical University of Denmark* (2008).

[55] Mathew Hall et al. *Moorpy (quasi-static mooring analysis in python)*. Tech. rep. National Renewable Energy Lab.(NREL), Golden, CO (United States), 2021.

[56] Yingyi Liu. 'HAMS: A frequency-domain preprocessor for wave-structure interactions—Theory, development, and application'. In: *Journal of Marine Science and Engineering* 7.3 (2019), p. 81.

[57] SA Ning. *CCBlade Documentation: Release 0.1. 0*. Tech. rep. National Renewable Energy Lab.(NREL), Golden, CO (United States), 2013.

[58] N. Abbas et al. 'A Reference Open-Source Controller for Fixed and Floating Offshore Wind Turbines'. In: *Wind Energy Science Discussions* 2021 (2021), pp. 1–33. DOI: 10.5194/wes-2021-19. URL: https://wes.copernicus.org/preprints/wes-2021-19/.

[59] [ENV]. '1993-1-6: Eurocode 3–design of steel structures–part 1.6: strength and stability of shell structures'. In: (2007).

[60] *Trading economics, prices of commodities*. URL: https://tradingeconomics.com/commodity (visited on 12th May 2023).

[61] *Stevpris MK5 website*. URL: https://delmarsystems.com/products/anchors/stevpris-mk5/ (visited on 12th May 2023).

[62] K. Deb et al. 'A fast and elitist multiobjective genetic algorithm: NSGA-II'. In: *IEEE Transactions on Evolutionary Computation* 6.2 (2002), pp. 182–197. DOI: 10.1109/4235.996017.

[63] MD McKay. 'McKay, MD, Conover, WJ and RJ Beckman (1979)'. In: *A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. Technometrics* 21 (1979), pp. 239–245.

[64] Willard J Pierson Jr and Lionel Moskowitz. 'A proposed spectral form for fully developed wind seas based on the similarity theory of SA Kitaigorodskii'. In: *Journal of geophysical research* 69.24 (1964), pp. 5181–5190.

[65] DNV GL. *DNVGL-ST-0437: Loads and site conditions for wind turbines*. 2016.

[66] Trevor Hastie et al. *The elements of statistical learning: data mining, inference, and prediction*. Vol. 2. Springer, 2009.

[67] James P Barrett. 'The coefficient of determination—some limitations'. In: *The American Statistician* 28.1 (1974), pp. 19–20.

[68] *Sklearn module: Linear regression.* (Visited on 29th July 2023).

[69] *Sklearn module: Decision trees.* URL: https://scikit-learn.org/stable/modules/generated/sklearn.tree.DecisionTreeClassifier.html (visited on 29th July 2023).

[70] *Sklearn module: Random forest.* URL: https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html (visited on 29th July 2023).

[71] *Xgb module: Default hyperparameters.* URL: https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LinearRegression.htm (visited on 29th July 2023).

Appendix

A IEA 15 MW reference wind turbine

IEA 15-MW reference wind turbine		
Parameter	Units	Value
Power rating	MW	15
Turbine class	-	IEC Class 1B
Specific rating	W/m^2	332
Rotor orientation	-	Upwind
Number of blades	-	3
Control		Variable speed Collective pitch
Cut-in wind speed	m/s	3
Rated wind speed	m/s	10.59
Cut-out wind speed	m/s	25
Design tip-speed ratio	-	9.0
Minimum rotor speed	rpm	5.0
Maximum rotor speed	m/s	95
Rotor diameter	m	240
Airfoil series	-	FFA-W3
Hub heihgt	m	150
Hub overhang	m	11.35
Rotor precone angle	deg	-4.0
Blade prebend	m	4
Blade mass	t	65
Drivetrain	-	Direct drive
Shaft tilt angle	deg	6
Rotor nacelle assembly mass	t	1017
Transition piece height	m	15
Monopile embedment length	m	45
Monopile base diameter	m	10

Table 15: IEA 15 *MW* reference wind turbine [14]

B Reference design geometry
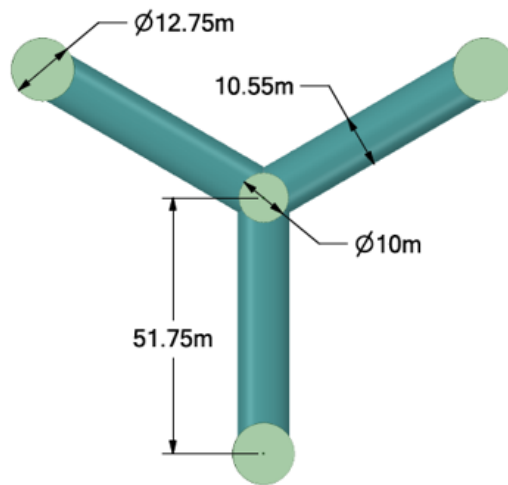
Figure 63: Reference design geometrical details: top view
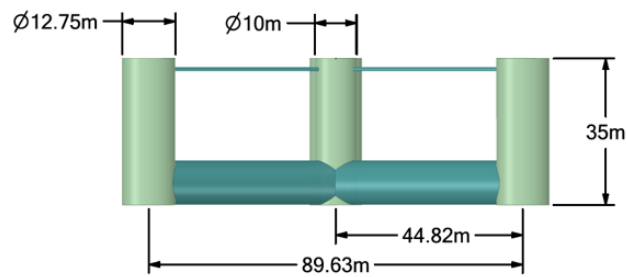
Figure 64: Reference design geometrical details: front view
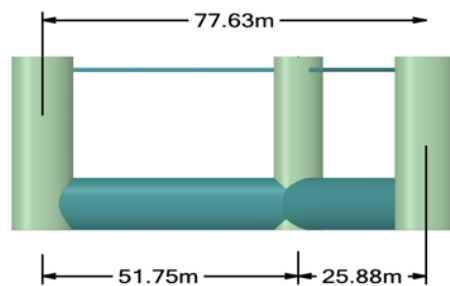
Figure 65: Reference design geometrical details: side view

C Reference design and VolturnUS-S added mass comparison
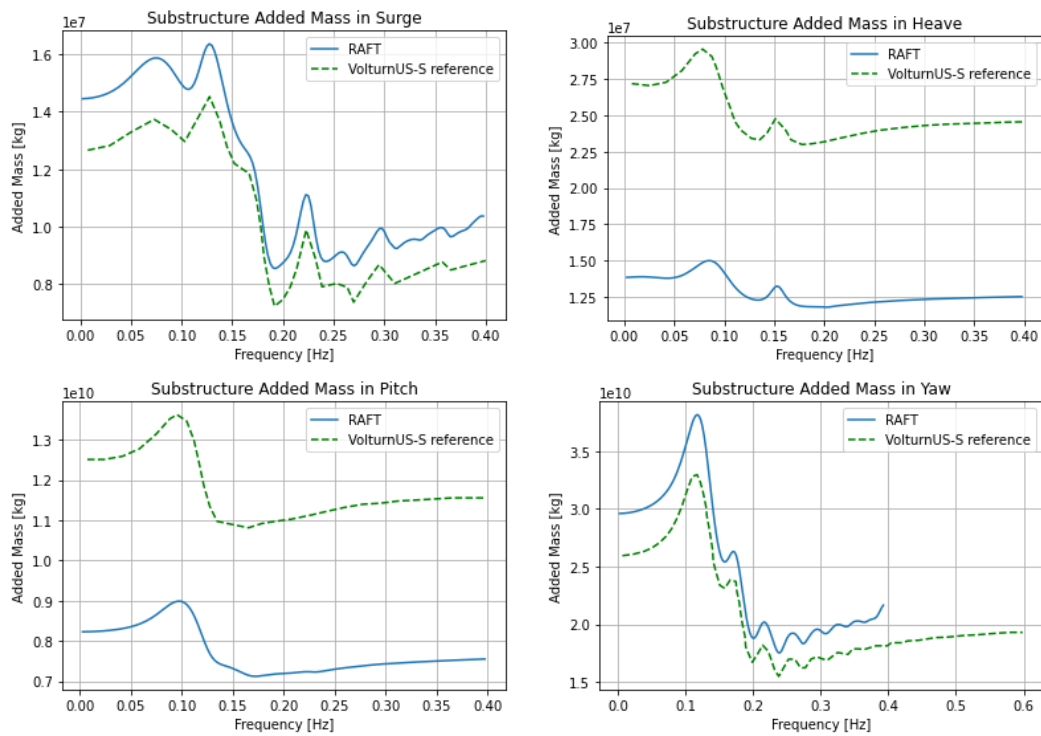
Figure 66: Added mass comparison between the VolturnUS-S and the alternative design. Reference data based on the VolturnUS-S report. [33]

.

D   RAFT-ROSCO coupling

The diagram of Figure 67 gives an overview of the process followed to integrate the controller tuning into RAFT. The controller's tuning is conducted between the natural frequency calculation and the static and dynamic analysis of RAFT, as mentioned in Section 4.3. To tune the controller of the wind turbine for a new design configuration, the following steps are taken:

- For every new design configuration, the eigenvalues and natural frequencies of the system are calculated.

- The pitch natural frequency is imported into the ROSCO, and the algorithm conducts the tuning process.

- ROSCO tunes the controller, reporting the torque and pitch control PI gains, together with the floating gain. All of these outputs are functions of wind speed. The blade's pitch angle is also computed by ROSCO as a function of wind speed.

- The pitch and floating gains are translated from the function of wind speed to functions of pitch angle, based on the pitch-wind speed relation reported by ROSCO. This process makes those outputs compatible with RAFT's algorithm.

- The tuned controller configurations are overwritten into the YAML RAFT file, followed by the static and dynamic analysis.
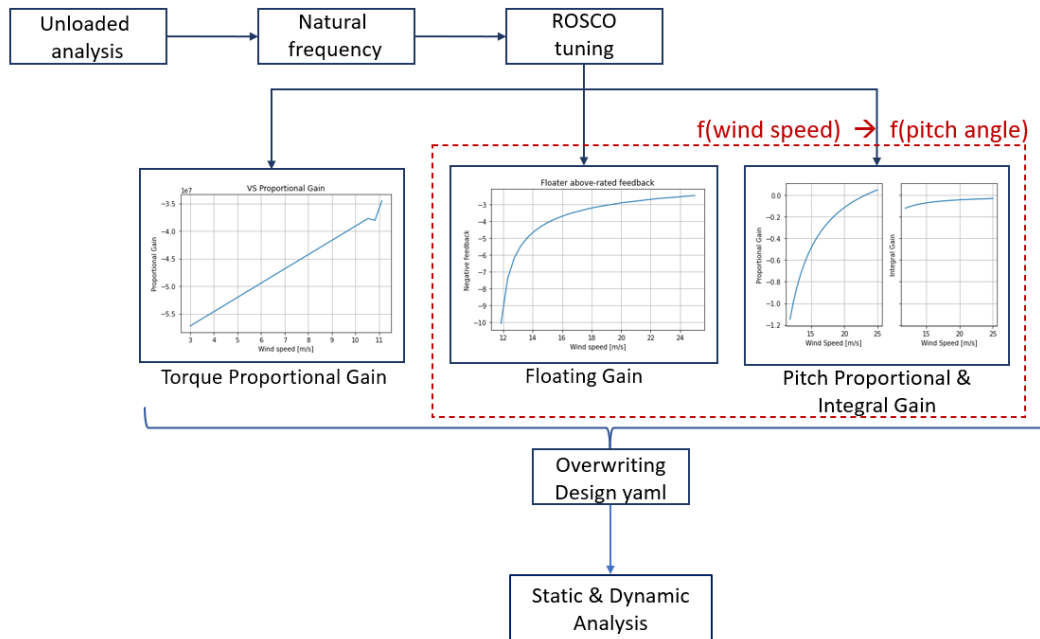
Figure 67: RAFT and ROSCO coupling overview.

E   Geometry generator

A script has been developed to generate new substructure geometries, according to the design variables. The geometry YAML files, that RAFT processes, have a very strict structure, increasing the possibility of human error. For that reason, a Python script is made, able to import the reference YAML file for the VolturnUS-S model, modify it properly, and export the new YAML file. The steps followed by the script are to create a copy of the reference YAML file, so that the original is not lost, and then modify the copy based on the variables affected by a specific set of $[Lo, Do]$. This simple process is depicted in Figure 68. In addition, this code can overwrite the existing updated geometry with a new one, making it able to be implemented inside a loop.

Figure 68: Geometry modifier script process.

F   Wall thickness calculation additional formulas

Calculating α	
α_χ	$\alpha_\chi = \frac{0.62}{1+1.9(\Delta_{wk}/t_{pon})^{1.44}}$ with $\Delta_{wk} = \frac{1}{Q_\chi} \cdot \sqrt{\frac{r}{t_{pon}}}$ with $Q_\chi = 16$
α_τ	$\alpha_\tau = 0.5$
α_θ	$\alpha_\theta = 0.5$

Table 16: Calculating the elastic imperfection reduction factor, α.

Calculating λ			
	λ_χ	λ_τ	λ_θ
$\bar{\lambda}$	$\bar{\lambda}_\chi = \frac{f_{yk}}{\sigma_{\chi_{Rcr}}}$	$\bar{\lambda}_\tau = \frac{f_{yk}\sqrt{3}}{\tau_{\theta_{Rcr}}}$	$\bar{\lambda}_\theta = \frac{f_{yk}}{\theta_{Rcr}}$
$\bar{\lambda}_p$	$\bar{\lambda}_{p,\chi} = \sqrt{\frac{\alpha_\chi}{1-\beta}}$	$\bar{\lambda}_{p,\tau} = \sqrt{\frac{\alpha_\tau}{1-\beta}}$	$\bar{\lambda}_{p,\theta} = \sqrt{\frac{\alpha_\theta}{1-\beta}}$
λ_o	$\lambda_{o,\chi} = 0.2$	$\lambda_{o,\chi} = 0.4$	$\lambda_{o,\chi} = 0.4$

Table 17: Calculating the shell slenderness parameter, $\bar{\lambda}$.

Calculating $\tau_{\theta,Rcr}$	
ω	$\omega = \frac{L_{pon}}{\sqrt{r_{pon} \cdot t_{pon}}}$
C_{tau}	$C_\tau = \begin{cases} \sqrt{1 + \frac{42}{\omega^3}}, & \text{if } \omega \leq 10 \\ \frac{1}{3}\sqrt{\frac{\omega \cdot t_{pon}}{r_{pon}}}, & \text{if } \omega \geq 8.7\frac{r_{pon}}{t_{pon}} \\ 1, & \text{if } 10 \leq \omega \leq 8.7\frac{r_{pon}}{t_{pon}} \end{cases}$
$\tau_{\theta,Rcr}$	$\tau_{\theta,Rcr} = 0.75 \cdot E \cdot C_\tau \cdot (t_{pon}/r_{pon}) \cdot \sqrt{1/\omega}$

Table 18: Calculating the elastic critical shear buckling stress, $\tau_{\theta,Rcr}$.

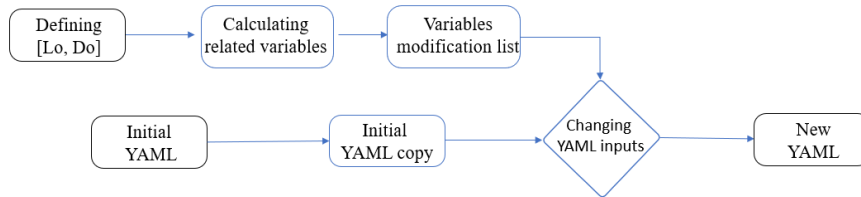| Calculating $\sigma_{\chi,Rcr}$	
ω	$\omega = \frac{L_{pon}}{\sqrt{r_{pon} \cdot t_{pon}}}$
C_χ	$C_x = \begin{cases} 1.36 - (1.83/\omega) + 2.07/\omega^3, & \text{if } \omega \leq 1.7 \\ 3, & \text{if } \omega \geq \frac{0.5r_{pon}}{t_{pon}} \\ 1, & \text{if } 1.7 \leq \omega \leq \frac{0.5r_{pon}}{t_{pon}} \end{cases}$
$\sigma_{\chi,Rcr}$	$\sigma_{\chi,Rcr} = \frac{0.605 \cdot E \cdot C_\chi \cdot t_{pon}}{r_{pon}}$

Table 19: Calculating the elastic critical meridional buckling stress, $\sigma_{\chi,Rcr}$.

Calculating $\sigma_{\theta,Rcr}$	
ω	$\omega = \frac{L_{pon}}{\sqrt{r_{pon} \cdot t_{pon}}}$
$C_{\theta,s}$	$C_{\theta,s} = 0.6 + \frac{1}{\omega^2} - \frac{0.3}{\omega^3}$ with $C_\theta = 0.6$
α_χ	$\sigma_{\theta,Rcr} = \begin{cases} 0.92 \cdot E \cdot \frac{C_{\theta,s}}{\omega} \cdot \frac{t_{pon}}{r_{pon}}, & \text{if } \frac{\omega}{C_{theta}} \leq 20 \\ E \cdot (\frac{t_{pon}}{r_{pon}})^2 \cdot (0.275 + 2.03(C_\theta \cdot r_{pon}\frac{t_{pon}}{\omega})^4, & \text{if } \frac{\omega}{C_{theta}} \geq 1.63\frac{r_{pon}}{t_{pon}} \\ 0.92 \cdot E \cdot \frac{C_\theta \cdot t_{pon}}{\omega \cdot r_{pon}}, & \text{if } 20 \leq \frac{\omega}{C_{theta}} \leq 1.63\frac{r_{pon}}{t_{pon}} \end{cases}$

Table 20: Calculating the elastic critical circumferential buckling stress, $\sigma_{\theta,Rcr}$.

Wspd m/s	Significant wave height [m]										
	0-1	1-2	2-3	3-4	4-5	5-6	6-7	7-8	8-9	9-10	Sum
0.5	391	208	28	2	0	0	0	0	0	0	629
1.5	1161	649	77	2	1	0	0	0	0	0	1890
2.5	1899	1014	122	4	0	0	0	0	0	0	3039
3.5	2260	1427	170	17	0	0	0	0	0	0	3874
4.5	2586	1799	263	22	2	0	0	0	0	0	4672
5.5	2632	2268	333	25	3	0	0	0	0	0	5261
6.5	2668	2876	421	51	3	0	0	0	0	0	6019
7.5	2374	3422	604	63	5	2	0	0	0	0	6470
8.5	1880	4018	862	78	5	0	0	0	0	0	6843
9.5	1097	4527	1014	144	12	0	0	0	0	0	6794
10.5	593	4634	1397	217	17	0	0	0	0	0	6858
11.5	292	4029	1789	333	24	5	1	0	0	0	6473
12.5	84	3309	2126	508	58	3	0	0	0	0	6088
13.5	35	2264	2355	673	88	5	0	0	0	0	5420
14.5	13	1214	2447	746	163	12	0	0	0	0	4595
15.5	4	509	2282	868	256	26	0	0	0	0	3945
16.5	0	170	1587	897	290	48	2	1	0	0	2995
17.5	0	38	959	969	342	83	8	1	0	0	2400
18.5	0	14	490	922	353	122	23	0	0	0	1924
19.5	0	0	166	747	342	170	30	2	0	0	1457
20.5	0	0	52	466	351	145	41	7	0	0	1062
21.5	0	1	9	195	302	125	42	12	1	0	687
22.5	0	1	4	72	232	105	54	8	4	0	480
23.5	0	0	4	22	130	86	40	16	5	0	303
24.5	0	0	0	7	38	86	38	12	1	0	182
25.5	0	0	0	2	12	48	29	9	7	1	108
26.5	0	0	0	1	9	22	17	9	5	1	64
27.5	0	0	0	0	5	5	12	2	0	0	24
28.5	0	0	0	0	2	2	3	3	0	0	10
29.5	0	0	0	0	1	2	0	6	1	0	10
30.5	0	0	0	0	0	0	2	1	0	0	3
31.5	0	0	0	0	0	0	0	0	1	0	1
32.5	0	0	0	0	0	0	0	0	2	0	2
33.5	0	0	0	0	0	0	0	0	1	0	1
34.5	0	0	0	0	0	1	0	0	0	0	1
Sum	19969	38391	19561	8053	3046	1103	342	89	28	2	90584
Mean	5.64	8.99	12.7	15.91	18.52	20.64	22.43	23.99	25.74	26.15	
Std	2.59	3.31	3.52	3.39	3.2	2.97	2.72	2.87	3.19	0.65	

Figure 69: Buchan Deep scatter table of wind speed versus significant wave height.

	Wave period (T) – (s)														
H (m)	**2 -3**	**3 – 4**	**4 – 5**	**5 – 6**	**6 – 7**	**7 – 8**	**8 – 9**	**9 – 10**	**10 – 11**	**11 – 12**	**12 - 13**	**13 - 14**	**14 - 15**	**15 - 16**	**16 - 17**
0 – 1	51931693	43368463	38977815	27254815	16532061	8995690	4519463	2163441	1013690	471752	219090	101746	47361	22155	10440
1 – 2		5642200	12948068	18254008	16850210	11673876	6762466	3544924	1727220	793291	349402	150011	63527	26735	11239
2 – 3		347338	1545511	3408237	4518801	4178532	3014359	1817695	957252	458454	204714	86102	34371	13149	4869
3 - 4			185265	634845	1211106	1441761	1228192	819609	459094	227537	102793	43192	17155	6492	2348
4 – 5				133036	298391	462108	475775	359927	216200	109478	48911	20038	7733	2858	1023
5 – 6				20535	70966	144039	180184	155848	101606	53393	23904	9552	3530	1242	426
6 – 7				3297	16345	43015	65739	65702	47138	26156	11941	4725	1693	570	186
7 – 8					4084	12310	23161	26982	21535	12779	6033	2401	847	276	86
8 – 9					835	3408	7900	10794	6200	3063	1242	437	140	42	
9 – 10						1080	2610	4209	4265	2981	1555	649	230	73	22
10 - 11						269	838	1602	1847	1417	787	341	123	39	11
11 - 12						66	262	596	785	666	397	180	66	21	6
12 - 13							96	217	328	309	198	95	36	12	3
13 - 14							28	77	135	141	98	50	20	6	2
14 - 15							8	27	54	64	48	26	11	4	1
15 - 16							2	9	22	28	24	13	6	2	1
16 - 17								4	8	12	11	7	3	1	
17 - 18								1	3	5	5	4	2	1	
18 - 19									1	2	3	2	1		
19 - 20											1	1	1		
20 - 21											1				

	Wave period (T) – (s)														
H (m)	**17 – 18**	**18 – 19**	**19 – 20**	**20 – 21**	**21 – 22**	**22 – 23**	**23 – 24**	**24 – 25**	**25 – 26**	**26 – 27**	**27 – 28**	**28 - 29**	**29 - 30**	**30 - 31**	**Sum**
0 – 1	4960	2370	1134	540	255	119	54	24	11	4	2	1			**195639149**
1 – 2	4734	1997	838	348	141	56	22	8	3	1					**78805325**
2 – 3	1760	623	215	73	24	8	3	1							**20592091**
3 - 4	814	271	86	26	8	2	1								**6380597**
4 – 5	356	121	40	13	4	1									**2136013**
5 – 6	144	49	16	5	2	1									**765442**
6 – 7	60	19	6	2	1										**286595**
7 – 8	26	8	3	1											**110532**
8 – 9	12	4	1												**43748**

Figure 70: Buchan Deep scatter table of significant wave height versus wave period.
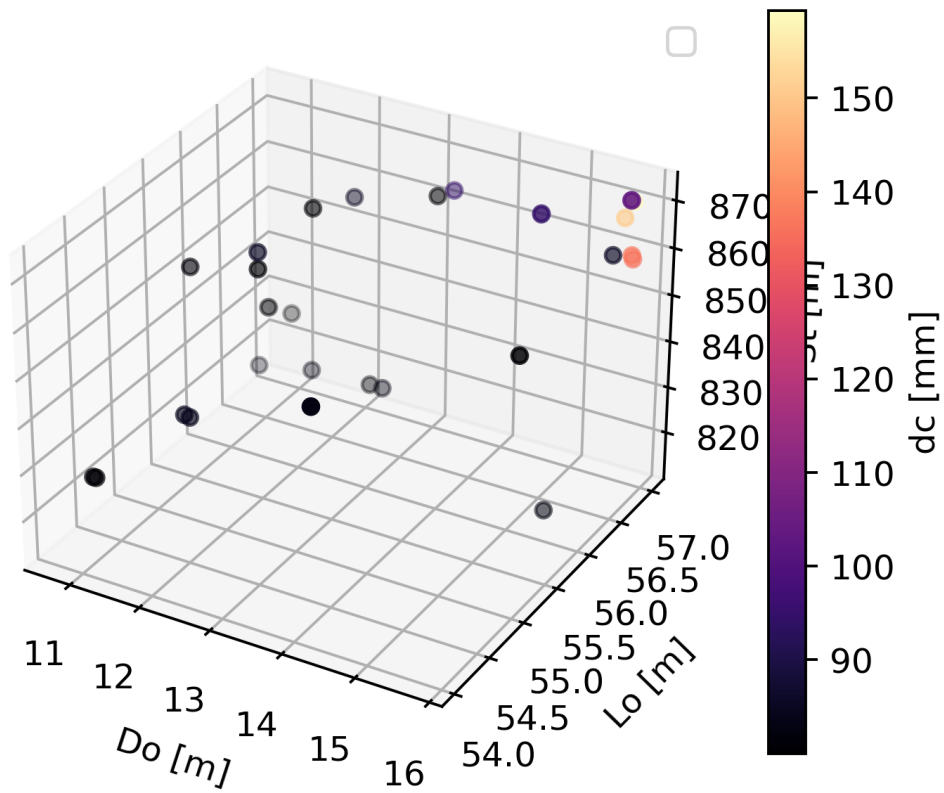
H   RAFT-based Pareto front population

Figure 71: RAFT-based Pareto (optimized) population.

I Additional graphs

A RAFT Floater variations contour plots

In order to produce clear contour plots that demonstrate the characteristics of the design space, the colour bars are bounded in specific cases. For the standard deviation of power, σ_P, the colour bar is bounded to a maximum of 1.07 MW. For maximum surge, the colour bar is bounded to a maximum of 27.5 m. For maximum pitch, the colour bar is bounded to a maximum of 14deg. Finally, for the maximum mooring lines' tension, the colour bar is bounded to a maximum of 3.6 MN.
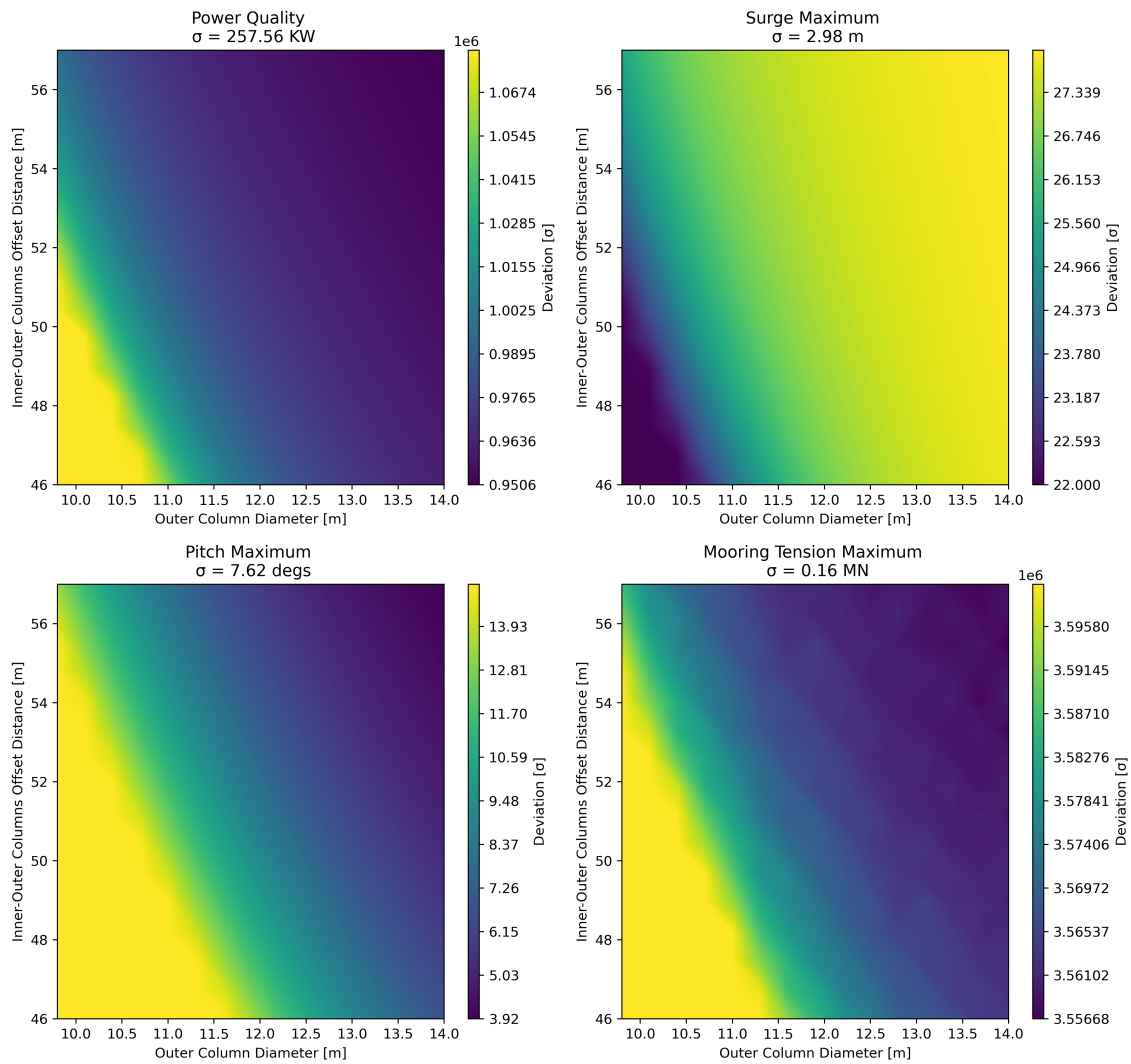

Figure 72: RAFT outputs contour plots, for variations of the floater design variables.

B RAFT Mooring variations contour plots

In order to produce clear contour plots that demonstrate the characteristics of the design space, the colour bars are bounded in specific cases. For the standard deviation of power, σ_P, the colour bar is bounded to a minimum of 1.03 MW. For maximum surge, the colour bar is bounded to a minimum of 17.0 m. Finally, for the maximum mooring lines' tension, the colour bar is bounded to a maximum of 3.7 MN.
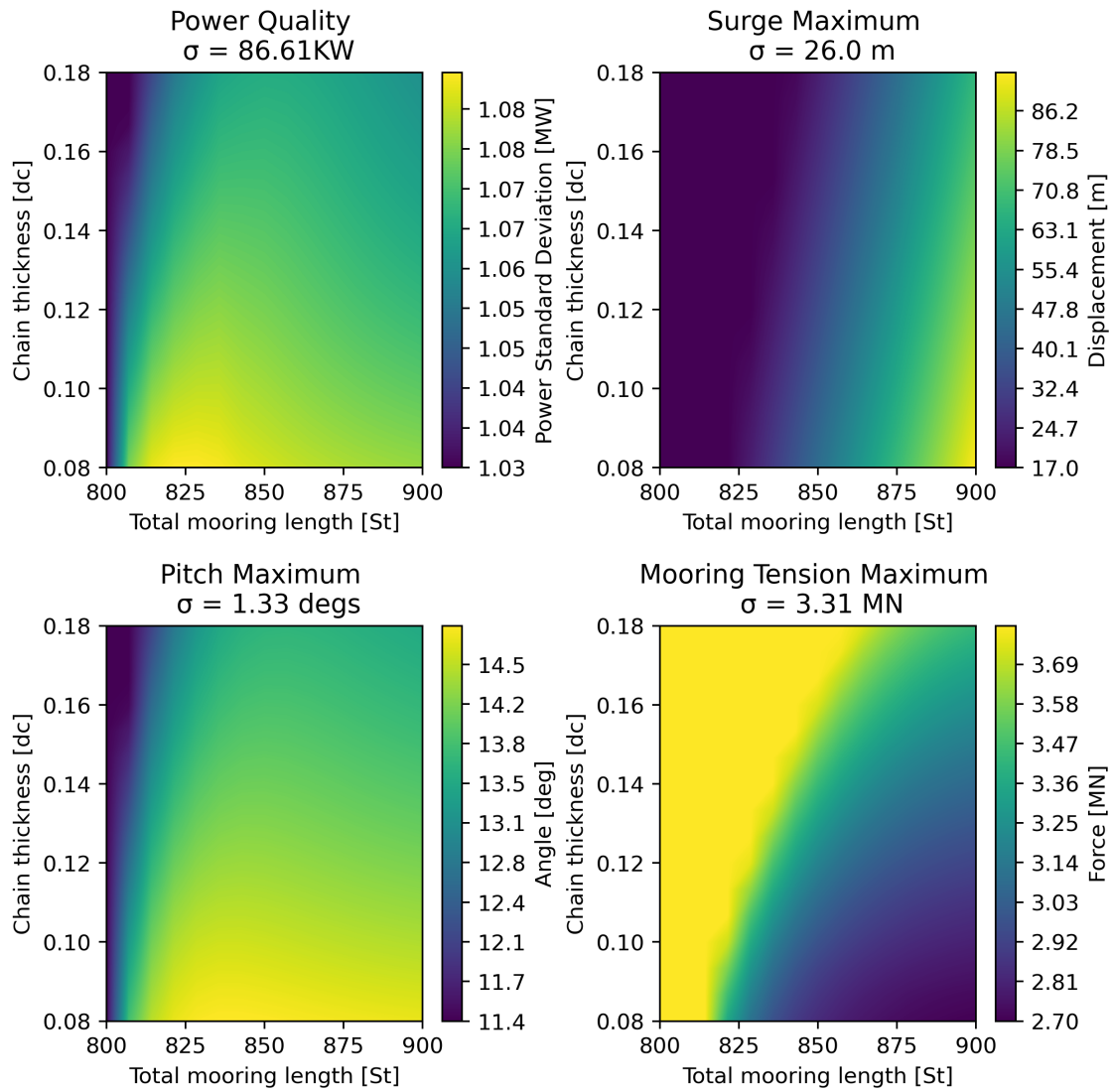

Figure 73: RAFT outputs contour plots, for variations of the mooring design variables.

J Surrogate models hyperparameters

In the following tables, the hyperparameters of each surrogate model examined can be found. Based on their values, the results mentioned in Section 5.1.1, Section 4.11 and Figure 50 can be reproduced. For the rest of the hyperparameters included inside each method, the default value has been utilized.

Linear Regression			
	PyHAMS	Operational End-to-End	Extreme End-to-End
Hyperparameters	-		
Default hyperparameters	sklearn module, linear regression [68]		

Table 21: Hyperparameters utilized for the linear regression surrogate model.

Decision Trees			
	PyHAMS	Operational End-to-End	Extreme End-to-End
Hyperparameters	max depth: None		
Default hyperparameters	sklearn module, decision trees [69]		

Table 22: Hyperparameters utilized for the decision trees surrogate model.

Random Forest			
	PyHAMS	Operational End-to-End	Extreme End-to-End
Hyperparameters	Number of estimators: 200, max depth: None		
Default hyperparameters	sklearn module, random forest [70]		

Table 23: Hyperparameters utilized for the random forest surrogate model.

Extreme Gradient Boosting			
	PyHAMS	Operational End-to-End	Extreme End-to-End
Hyperparameters	Objective: reg:squarederror, Learning rate: 0.1 max depth: None, number of estimators: 200		
Default values	xgboost module [71]		

Table 24: Hyperparameters utilized for the extreme gradient boosting surrogate model.

Artificial Neural Networks		
	Operational End-to-End	Extreme End-to-End
Structure	Layer 1: 7 neurons, 'relu' activation Layer 2: 14 neurons, 'relu' activation Layer 3: 28 neurons, 'relu' activation Layer 4: 14 neurons, 'relu' activation Layer 5: 1 neuron, 'linear' activation	Layer 1: 4 neurons, 'relu' activation Layer 2: 8 neurons, 'relu' activation Layer 3: 16 neurons, 'relu' activation Layer 4: 16 neurons, 'relu' activation Layer 5: 8 neurons, 'relu' activation Layer 6: 1 neuron, 'linear' activation
Hyperparameters	epochs: 1800, batch size: 60	epochs: 600, batch size: 20

Table 25: Hyperparameters utilized for the artificial neural networks surrogate model.