MSc Thesis

An improved, coupled frequency domain model for floating offshore wind turbines including aerodynamics, hydrodynamics and control

MT54035

M.J. Schipper

Delft University of Technology

TU Delft

MARIN

MSc Thesis

An improved, coupled frequency domain model for floating offshore wind turbines including aerodynamics, hydrodynamics and control

by

M.J. Schipper

performed at

MARIN

to obtain the degree of Master of Science in Marine Technology in the specialization of Ship Hydromechanics at the Delft University of Technology,
to be defended publicly on Monday November 27, 2023 at 01:30 PM.

Student number:	4372344	
Thesis number:	MT.23/24.013.M	
Thesis committee:	ir. G.J. Bouman,	MARIN, daily supervisor
	Dr.ir. P. Naaijen,	TU Delft, supervisor
	Dr.ing. S. Schreier,	TU Delft, chairman
	Prof.dr.ir. J.W. van Wingerden,	TU Delft, staff member

Cover: An AI generated ”Floating Offshore Wind Turbine with three blades in wavey conditions during sunset, using a Vincent van Gogh painting style” (Nightcafé)

TU Delft Delft University of Technology

MARIN

Preface

Komt wel goed

Voor jullie is dit het eerste wat jullie lezen, voor mij het laatste wat ik type aan mijn thesis. Het einde van een tijdperk. Een leuk tijdperk, wel te verstaan. Ik ben nooit met tegenzin richting Delft gegaan. Oké, toegegeven, soms wel door de vroege wekker, maar niet vanwege de studie. Dit heeft veel te maken met de leuke mensen die ik daar om me heen heb gehad. Elke dag gelachen, en ik moet bijvoorbeeld om 12:00 uur nog altijd denken aan curve fever. Dank allemaal voor de leuke en leerzame tijd.

Dat ik me altijd goed heb gevoeld tijdens mijn studie komt niet alleen maar door de omgeving in Delft, maar vooral ook door de mensen thuis. Natuurlijk bedank je dan je familie, dus papa, mama, Jasper, alle katten, ontzettend bedankt voor het warme nest thuis. We boffen ontzettend met elkaar. Dat geldt natuurlijk ook voor jullie, Gerry, Nilüfer, en ook Spikey. Aslan Jurry is ontzettend blij met jullie als schoonfamilie. Nu ga ik verder niet mijn hele stamboom en vriendenkring hier opsommen, maar ja, het geldt oprecht ook voor jullie allemaal. Ik word altijd vrolijk van jullie aanwezigheid.

Nu denkt de oplettende lezer: ”Hé, er mist er nog eentje!”. Vrees niet, ik heb de beste tot het laatst bewaard. Ekkie, duizend maal dank voor het zijn wie je bent. Ik ben een ontzettende geluksvogel met jou aan mijn zijde. De afgelopen weken kreeg ik steeds meer plannen te horen wat we samen konden gaan doen als ik klaar was, en ik kijk er nu al naar uit om ze stuk voor stuk uit te voeren. Ik heb zin in onze toekomst samen!

Nu zijn er nog een paar die ik wil bedanken voor de afgelopen periode, beginnend met Gijs. Allereerst bedankt voor de kans om af te studeren op dit interessante onderwerp (ook jij bedankt daarvoor Erik-Jan). Ik geloof erin dat de toekomst ”groen” kan zijn met behulp van technische ontwikkelingen, en ik ben dankbaar dat ik er op deze manier een klein steentje aan heb kunnen bijdragen. Maar bovenal wil ik je bedanken voor je begeleiding gedurende het project. Ik heb veel van je geleerd, en de gezellige thee (nee, geen koffie voor mij) momentjes op kantoor samen ga ik zeker missen. Wij komen elkaar nog wel weer tegen in de toekomst :)

Niet alleen de begeleiding van jou is me goed bevallen, ook de werksfeer bij MARIN in zijn geheel. Wim, Robin, Joep, en iedereen die ik een vraag heb gesteld, bedankt voor de tijd die jullie hebben genomen om hem te beantwoorden.

And last but not least for my time at MARIN, my fellow interns. Thank you all for the good times we had. I already miss our ping pong tournaments (also thank you Naman for guiding us to the ”big” ping pong table) and the joy we had during the lunch breaks, or outside office hours. Together we survived the cold temperatures at the office, hehe.

Peter, ook jij natuurlijk bedankt dat je de tijd wilde nemen om me namens de TU te begeleiden gedurende dit project!

Tenslotte zag ik dat andere mensen een inspirerende quote in hun thesis verwerkten. Nu kon ik niet kiezen welke beter was, dus laat ik het aan de lezer om zijn of haar favoriet te kiezen. Succes met de keuze, en veel plezier met het lezen van mijn thesis! Doei.

”Nature is the source of all true knowledge”
- Leonardo Da Vinci

”Komt wel goed”
- Jurrien Schipper

M.J. Schipper
Utrecht, November 2023

Summary

Renewable energy is fundamental for limiting the global temperature increase. Policies to limit this increase have set out a global electricity generation mix wherein wind and solar will become the main sources in the next three decades, with wind generating a slightly higher overall share of global electricity and thus producing more electricity than any other energy source. Although the renewable market is growing fast, an increase of thirteen times the annual deployment of offshore wind capacity compared to the current rates is needed by 2030.The ambitious growth plans lead to some important limitations faced by bottom fixed offshore wind turbines (OWTs). this technology will reach a limit in how much it can be utilized, since the suitable areas in shallow waters are becoming more scarce over time. As a result, up to 80% of the wind resource's potential can not be harnessed using fixed renewables. Floating Offshore Wind Turbines (FOWTs) therefore provide an option to unlock the enormous potential that exists around the world, with the extra perk that they are often located farther offshore where they benefit from stronger and more consistent winds. The FOWT technology however is still in their preliminary stage of technology readiness. Hitherto, already more than 30 FOWT concepts have been proposed, and the market still did not merge to one most favourable design to be optimized. Therefore the industry would benefit from cost efficient tools to evaluate FOWT designs, with the turbine and floater included in one and the same tool. This Thesis aims to develop such a coupled and cost efficient tool.

Modelling of FOWTs is challenging due to the strong coupling between the aerodynamics of the turbine and the hydrodynamics of the floating structure and mooring system, which makes the dynamic response of the whole system highly complex. A variety of numerical and physical modelling approaches can be used to determine this response.

The frequency-domain method (FD) is chosen for further development. These methods are considered as efficient tools for analyzing first design iterations. Given the ongoing development of new concepts for FOWTs, the industry could benefit from such a tool. Currently, only one such open-sourced method is developed (RAFT), but the method still has room for improvements. Time-domain simulations (TD) will be used for validating the FD model.

To improve on RAFT, a python wrapper is built to use DIFFRAC and aNySIM fredom for the calculations of the hydrodynamics and RAFT's Rotor class for the aerodynamics. The latter needed some adjustments to include the nacelle velocity feedback gain. It is shown that applying BEM calculations for the hydrodynamics already improves the FOWT's motion predictions compared to the 2D strip-theory applied in RAFT, but that including the aerodynamic coefficients improves the model even further, especially for the pitch motion. Attempts have been made to also improve on the mooring line modelling, trying to take mooring line drag and inertia effects into account, which impact the mooring line tension predictions for higher frequencies. The mooring line tension predictions however, did not improve. Linearizing the response in aNySIM based on the velocity standard deviations turned out to be insufficient. Therefore, to develop the model further it is suggested to adopt a lumped-mass mooring line model to improve the mooring tension predictions. Furthermore, the low frequency range wind excitation must be added in the final solver, in order for the model to be able to predict the motions due to the wind excitation in this frequency range. The current model is not yet capable of including this frequency dependent excitation.

Contents

List of Figures

List of Tables

Nomenclature

All standard and frequently used nomenclature can be found in the tables below. On several occasions in this document a different or unique notation is used, for example if when a figure is directly retrieved from literature that maintains different notations. If this applies, the description of that particular symbol is described locally.

Abbreviations

Abbreviation	Definition
1P	Turbine rotational speed (same as Ω)
3P	Rotor blade passing frequency, or the frequency at which a blade pass by the tower for a three-bladed turbine
6P	Twice the blade passing frequency
AI	Artificial Intelligence
BC	Boundary Condition
BEM	Boundary Element Method
BEM*	Blade Element Momentum method
CFD	Computational Fluid Dynamics
DLC	Design Load Case
DOF	Degree of Freedom
EOM	Equations of Motion
FD	Frequency-domain
FFT	Fast Fourier Transform
FOWT	Floating Offshore Wind Turbine
GWEC	Global Wind Energy Council
IEA	International Energy Agency
JIP	Joint Industry Project
JONSWAP	Joint North Sea Wave Observation Project
LCOE	Levelized Cost of Energy
NREL	National Renewable Energy Laboratories, US
O&M	Operation and Maintenance
OWT	Offshore Wind Turbine
PSD	Power Spectral Density
QTF	Quadratic Transfer Function
RAFT	Response Amplitudes of Floating Turbines
RAO	Response amplitude operator
RHS	Right Hand Side
RNA	Rotor-nacelle assembly
ROSCO	Reference OpenSource Controller
SIL	Software In the Loop
TD	Time-domain
TLP	Tension Leg Platform
TSR	Tip Speed Ratio
VSVP	Variable Speed Variable Pitch

Symbols

Symbol	Definition
a	The axial component of the induced flow
a'	The tangential component of the induced flow
a_{ij}	Added mass coefficient in DOF i due to unitary motion in DOF j
$\mathbf{A}$	Matrix of all added mass coefficients
$\mathbf{A_{potential}}$	Matrix of added mass due to the radiation problem
$\mathbf{A_{morison}}$	Matrix of added mass for small elements
$\mathbf{A_{aero}}$	Matrix of aerodynamic added mass
A	Cross-sectional area
A_{wp}	Waterplane area
b_{ij}	Radiation damping coefficient in DOF i due to unitary motion in DOF j
b^v_{ij}	Viscous damping coefficient in DOF i due to unitary motion in DOF j
$\mathbf{B}$	Matrix of all damping coefficients
$\mathbf{B_{potential}}$	Matrix of damping due to the radiation problem
$\mathbf{B_{morison}}$	Matrix of viscous damping for small elements
$\mathbf{B_{aero}}$	Matrix of aerodynamic damping
c_{ij}	Restoring coefficient in DOF i due to unitary motion in DOF j
$\mathbf{C}$	Matrix of all restoring coefficients
$\mathbf{C_{hs,body}}$	Hydrostatic stiffness matrix
$\mathbf{C_{hs,moor}}$	Hydrostatic mooring matrix
C_d	Drag coefficient
C_L	Lift coefficient
C_m	Mass coefficient
D_r	Drag
E	Young's modulus
$\vec{F}$	Resulting force vector
$\vec{F}_{potential}$	Force vector due to the radiation problem
$\vec{F}_{drift}$	Force vector due to second order wave loads
$\vec{F}_{fk,morison}$	Froude-krylov force for small elements
$\vec{F}_{d,morison}$	Viscous drag loads for small elements
$\vec{F}_{aero}$	Turbulent wind excitation vector
g	Gravitational acceleration
H_h	Hub height
H_s	Significant wave height
I_{ij}	Moment of inertia in DOF i due to a unitary motion in DOF j
I_r	Rotor mass moment of inertia
k_{br}	Function of the maximum achievable power coefficient and the associated tip speed ratio
k_{Pi}	Proportional gain controlling parameter i
k_{Ii}	Integral gain controlling parameter i
L_i	Lift
L_F	Length, full scale
L_M	Length, model scale
m	Total mass of the FOWT
$\mathbf{M}$	Matrix of structural mass and inertia
N_g	Gear ratio
p	Pressure
P_{out}	Power output
Q	Aerodynamic torque
Q_g	Generator torque
R	Rotor radius
RMS	Root Mean Square value

Symbol	Definition
T	Thrust
T_x	Horizontal components of the tension
T_z	Vertical components of the tension
u	Flow velocity vector, represented by $\mathbf{u} = (u, v, w)^T$
U_w	Mean incoming wind speed
V	Floater's submerged volume
W	Relative wind speed
x_G	Location of centre of gravity in x direction
y_G	Location of centre of gravity in y direction
z_B	Location of centre of buoyancy in z direction
z_G	Location of centre of gravity in z direction
α	Angle of attack
β	Blade-pitch angle
ζ_a	Wave amplitude
η_1	Location of a discretized section
γ_r	Local tip speed ratio
λ	Scale ratio
λ'	Tip speed ratio associated with the maximum power coefficient
ν	Kinematic viscosity
ρ_a	Air density
ρ_w	Water density
τ_G	Electrical generator torque
Ψ	Angle of the relative wind speed
Φ	Linear potential function
Ω	Rotor speed

1

Introduction

1.1. Problem motivation

To tackle climate change and its negative impacts, world leaders agreed on the 2015 Paris Agreement. Renewable energy is fundamental for meeting the agreement's goal of limiting the global temperature increase to 1.5-2.0°C in 2100 compared with the pre-industrial levels. Policies to achieve these goals are proposed in e.g. the International Energy Agency's (IEA) Net Zero by 2050 Roadmap [55] and the IRENA's World Energy Transitions Outlook [56], with the latter visualized in figure 1.1. Both plans set out a global electricity generation mix wherein wind and solar will become the main sources in the next three decades, with wind generating a slightly higher overall share of global electricity and thus producing more electricity than any other energy source.

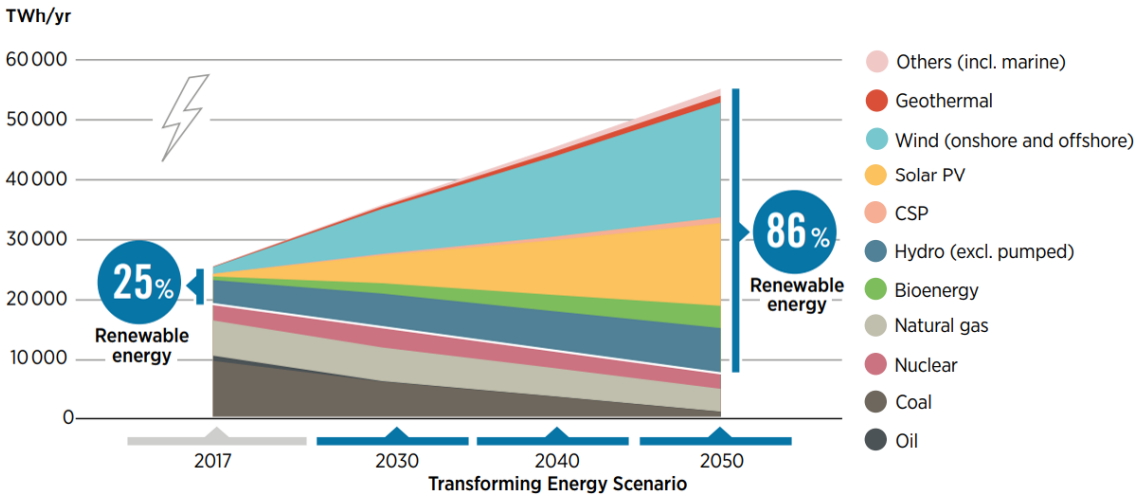

Figure 1.1: Breakdown of electricity generation by source from 2017 to 2050 as proposed by IRENA [56]

Although the renewable market is growing faster than the overall power demand [56], it is not growing nearly fast or widely enough to realise a secure and resilient global energy transition. For instance, the Global Wind Energy Council (GWEC) forecasts that by 2030, at the current rates of installation, the wind energy sector will have less than two-thirds of the capacity required for a 1.5°C and net zero pathway [37]. To meet up with the net zero pathway, an increase of thirteen times the annual deployment of offshore wind capacity compared to the current rates is needed by 2030 [36].

The ambitious growth plans lead to some important limitations faced by bottom fixed offshore wind turbines (OWTs). Since OWTs are only economically feasible in shallow waters, or explicitly up to 60 meters depth [106], this technology will reach a limit in how much it can be utilized, since the suitable areas in shallow waters still available to build OWTs are becoming more scarce over time [23]. As a result, up to 80% of the wind resource's potential can not be harnessed using fixed renewables [57].

Floating Offshore Wind Turbines (FOWTs) therefore provide an option to unlock the enormous potential that exists around the world, with the extra perk that they are often located farther offshore where they benefit from stronger and more consistent winds [74].

The FOWT technology however, is still in their preliminary stage of technology readiness [68]. So far, only a few real-world demonstration projects are installed. A good example of the opportunities that FOWTs have to offer is the Hywind Scotland park – a FOWT farm consisting out of five 6 MW Spar-type floating turbines [30] – as it has a capacity factor[1] of 52.6%, which is the highest for all offshore wind parks installed in the UK considering both bottom fixed and floating test sites, and well ahead of the 40% average [111]. On the other hand, one of the turbines of the Kincardine wind farm project – a FOWT farm of five 9.5MW semi-submersible-type floating turbines [95] – had to be towed back to port for a repair job due to a significant technical set back just months after it became fully operational [98], showing that the floating wind industry can face severe difficulties.

Besides showing the potential and possible challenges of FOWTs, these examples, currently being amongst the FOWT projects with the highest technology readiness level, also show the diversity in concepts being brought to the market. Hitherto, already more than 30 FOWT concepts have been proposed [68], and the market still did not merge to one most favourable design to be optimized. Knowledge from the oil- and gas industry, used in floater and mooring concepts, and knowledge from the fixed wind industry, used in rotor and drivetrain designs, cannot directly be implemented due to the FOWT's remarkable multidisciplinary character, making it a new engineering branch that faces its own challenges [23]. For example, the interaction between the platform motions and the wind turbine control system may result in serious instabilities [60, 67, 110].

The industry would therefore benefit from cost efficient tools to evaluate complete FOWT designs, since such a tool gives developers and designers the opportunity to evaluate a large amount of designs in a cost and time efficient manner. The Carbon Trust Floating Wind Joint Industry Project (JIP) [114] emphasizes that such tools need an improved interface between turbine suppliers and floater designers in order to have a more integrated design, since currently those components are developed rather separate from each other. Now, at the start of a project, clarity about how wind turbine and floater designers work together and couple their software is necessary. The development of universal coupling software could have a high potential to streamline this process [47]. This thesis aims to improve and validate such a coupled and cost efficient tool.

1.2. Thesis Outline

The problem motivation and the outline of this thesis are given in this chapter. To provide the reader with some knowledge regarding the different concepts that are most often adapted in FOWTs nowadays, chapter 2 will describe all the components present in a FOWT, their working principles, as well as providing examples.

The different possible modelling techniques that can be used to investigate the performance of a FOWT design will be described in chapter 3. One method will be chosen as the most interesting one to improve on, whose working principles will be explained in more detail thereafter. It is shown how each component, as described in the previous chapter, influences the equations of motion and how they are implemented in the chosen method. Once this is done, the state-of-the-art regarding the chosen method will be elaborated on, showing its workflow but also the shortcomings it still has, and thus its potential improvements.

Based on these shortcomings, the research question for this thesis is formed in chapter 4. Subquestions are set up that will help with answering the question. Furthermore some objectives are drafted that need to be achieved in order to answer the questions. A method to answer these questions and perform the objectives is thought of and elaborated on in chapter 5.

The results of this method will be verified using results of intermediate steps, and validated against a model with of higher fidelity. Based on these results, recommendations to improve the model are mentioned in chapter 6. Lastly, the thesis provides the conclusion in chapter 7.

[1] The ratio between actually produced energy divided by the installed capacity over a given period of time.

2

Description of a Floating Offshore Wind Turbine

The previous chapter showed the necessity of floating offshore wind turbines (FOWTs). This chapter describes the working principle of such a FOWT, by explaining how it generates its power and which components are needed to keep the system operating. The needed components can be distinguished into three main categories as follows: The floater, the mooring system, and the turbine, with the latter containing the tower and rotor-nacelle assembly (RNA). All the components are visualized in figure 2.1. The following sections describe their functionality, mention the different design ideas and refer to some projects where these design ideas are being adapted.

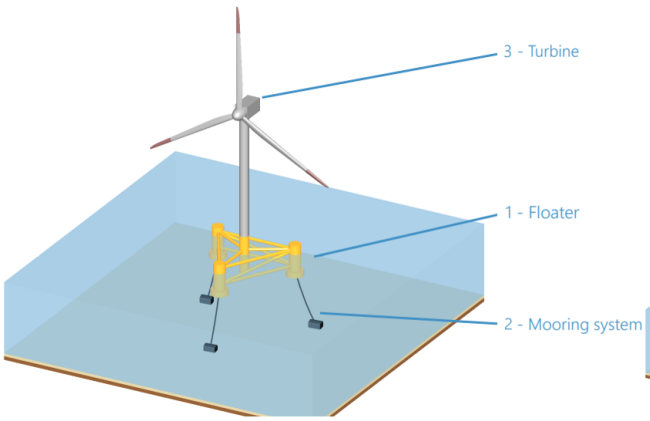

Figure 2.1: The three main components of a FOWT, with the tower situated between the floater and the turbine (Retrieved from [118])

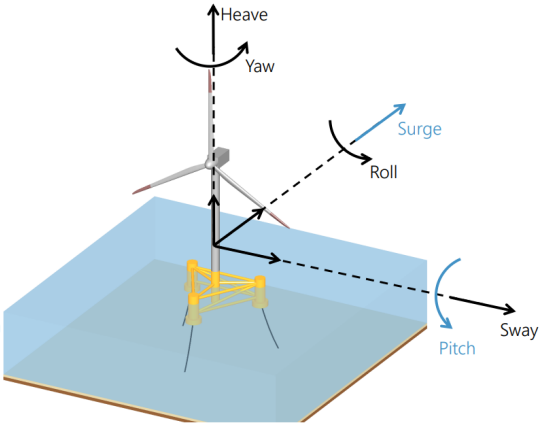

Figure 2.2: The six degrees of freedom for a FOWT (Retrieved from [118])

2.1. Floaters

The floater is the component that provides the carrying capacity for the turbine and part of the system's stability via buoyancy. The system has six degrees of freedom (DOFs): Three rigid body motions and three rotational motions, all presented in figure 2.2.

As mentioned in chapter 1, the FOWT technology is still evolving, trying to find more cost-effective design solutions. By 2018 already more than 30 FOWT concepts were proposed [68]. Most of them can be categorised into one of three main categories: Semi-submersibles, tension leg platforms (TLPs), and spars. Each category is depicted in figure 2.3 and are elaborated on in the upcoming subsections.

Semi-submersibles are buoyancy stabilized. They are large floating bodies that have a large second moment of area with respect to the rotational axis, so that it is stable due to a large restoring moment caused by buoyancy in case of a rotational displacement. This can either be due to a large waterplane

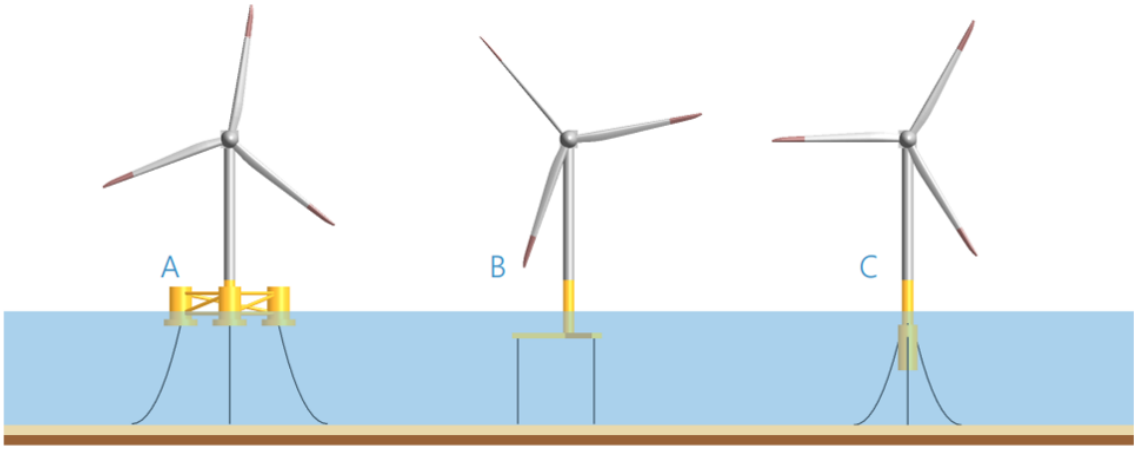

Figure 2.3: Three different types of floaters. A: Semi-submersible, B: TLP, C: Spar buoy (Retrieved from [118])

area, or with smaller cross-sectional areas at a distance from the system's central axis to generate a large moment of inertia. A smaller waterplane area reduces the wave excitation forces experienced by the floater. An example of a semi-submersible can be seen in figure 2.3 part A.

There are several open-sourced reference floaters available, designed for different turbine sizes and adapting different floater design ideas. For the semi-submersibles type there are the the OC4-DeepCwind [100] – designed for the 5 MW reference turbine [63] – and the the UMain VolturnUS-S [3] – designed for the 15 MW reference turbine [35]. Besides the reference floaters, the semi-submersible is also widely adapted in commercial projects [83]. Commercial examples are the Gusto-MSC Tri-Floater [43], Principle Power's WindFloat [96] and FLAGSHIP's design [34].

The self-stabilizing characteristic makes that they can be assembled in controlled environments, e.g. harbours, and can than be wet-towed to the desired location to be anchored to the seabed. However, the structure itself is relatively the most complex to construct. Furthermore they require the most material to be produced, meaning that the footprint of this design category is the highest.

Tension Leg Platforms (TLPs) are mooring stabilized as they rely on the mooring system to provide its stability. This system uses high tensioned mooring lines which pull the system down on several places far apart from each other, while the buoyancy of the system tries to pull the whole floater up, resulting in tight mooring lines and a much stiffer system in general compared to the other design categories. The mooring lines generate a restoring moment if the structure would be inclined and therefore provide the system's stability. An example of a TLP can be seen in figure 2.3 part B.

Examples of a commercial projects adapting the TLP design is Bluewater's TLP design [9] and SBM Offshore's Float4WindTM [103]. The main advantage of the TLP is the stiffness in heave and rotational motions so that the substructure's response characteristics are the closest to bottom-fixed foundations. Furthermore the substructure itself is a lightweight structure that is relatively simple to construct compared to the semi-submersible, and it has the smallest footprint out of the three design categories. However, the installation process, amongst others, is more challenging. This is because of the lack of wet-towing possibilities, thus onsite floating-to-floating installation is required at large water depths. Also, installing the mooring system with the high pretension is a complex and relatively expansive operation.

Spars are ballast stabilized. They have a large ballast weight deep at the bottom of the floating structure which moves the centre of gravity far below the centre of buoyancy, so that the system has a large stabilising righting moment in case of a rolling or pitching rotational displacement. An example of a Spar can be seen in figure 2.3 part C.

The Spar-type floater is also developed as an open-sourced model for the 5 MW reference turbine, which resulted in the OC3-Hywind design [59]. Although the Spar also has wet-towing possibilities, it requires evermore draught for higher rated turbines. Souza and Bachynski-Polić [113] designed a Spar-type supporting structure for a 20 MW turbine, which has a draught of approximately 100 meters. This makes the Spar-type less suitable for higher rated turbines if no deep water ports are available. If,

however, the turbine's size is not too large, the Spar has proven itself in e.g. Equinor's Hywind Scotland and Tampen projects [30, 31]. Furthermore the Spar-type floater is the easiest type to construct, and because it needs the weight they can be constructed out of e.g. concrete.

Other concepts are also possible, such as hybrid concepts (e.g. Stiesdal's Tetra design [116]), barges (e.g. BW Ideol's damping pool design [16]) and multi-turbine concepts (e.g. Hexicon's TwinWind design [50]). All these designs however, are closely related to one of the three main categories identified, or are a mix of these – in fact, all floating concepts are actually hybrid designs that gain static stability from all three main category ideas [19]. The design space regarding the floater's stability can therefore be represented as is done in figure 2.4, where the three main categories are the cornerstones of the stability triangle.

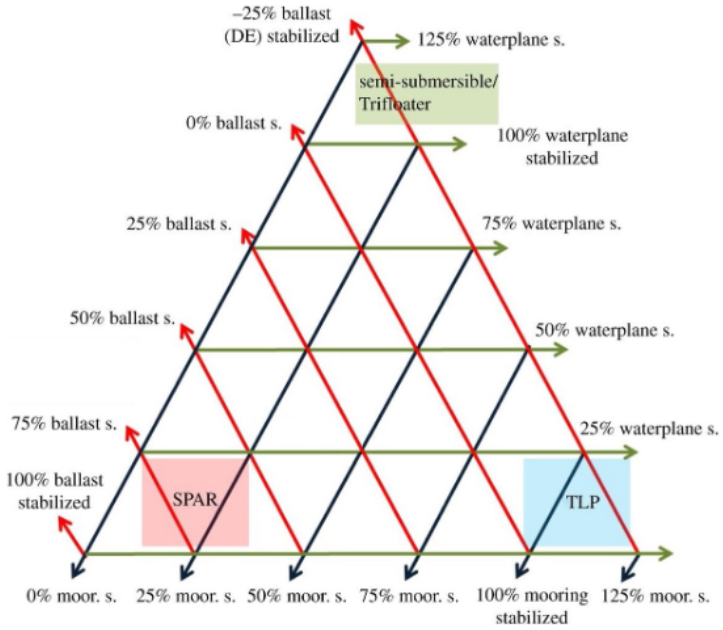

Figure 2.4: Stability triangle for floating structures (retrieved from [10])

2.2. Mooring Systems

The floater concepts as described in section 2.1 all just partly stabilize the system. Typically, the FOWT's stiffness in surge, sway and yaw direction depend on a mooring system. To prevent the floater from drifting away or yaw rotate freely, the substructures are embedded via mooring lines in the seabed. The options to embed the floater to the seabed can be categorized in three main categories, presented in figure 2.5. The costs of the mooring system depend on the category applied, but for all the categories holds that the costs are highly dependent on the water depth [14].

Catenary mooring relies on heavy chains that are lifted in case of an offset. The extra weight lifted will create a gravity based restoring force, but also a high drag. Because the anchor lines at the end lie horizontally on the seabed, the loads on them are also horizontal and relatively low. Therefore, cheap and easy to install anchors can be used. However, the footprint for such a system is the highest since the longest anchor lines, and thus the most material, are required [14]. To reduce this issue for wind farms, research is done in shared mooring systems [40, 69]. The catenary system, depicted in figure 2.3 part A, is the most widely used system, especially in shallow to medium-depth waters [72].

Taut mooring systems have no lines lying on the seabed in the static equilibrium position [72]. It relies on the anchor line tensile strength to generate the restoring force, as the lines are already under tension. However, the loads encountered by the anchor due to the stiff mooring lines are both horizontal and vertical resulting in the requirement of more expensive anchors [14]. The upside is that the system has

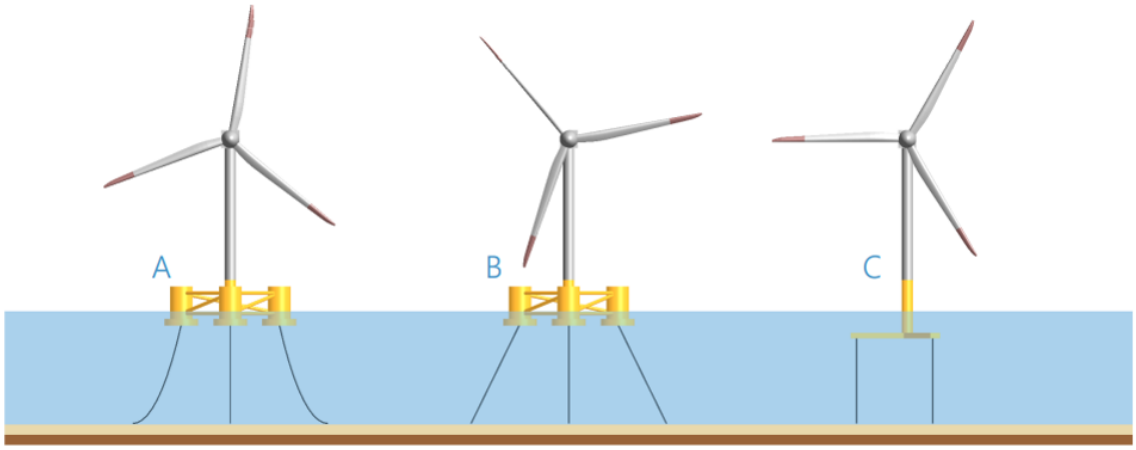

Figure 2.5: Three different types of mooring systems. A: Catenary mooring, B: Taut mooring, C: Tension leg (Retrieved from [118])

the highest stiffness in surge and sway direction and less material is needed compared to catenary mooring, resulting in a smaller footprint [118]. The taut mooring principle, as depicted in figure 2.3 part B, is more suitable for deep water applications, as it may be too stiff in shallow waters resulting in excessive line tensions [72].

Note that, just like was the case for the substructures, there are also hybrid designs possible. Mooring systems that have characteristics of both catenary and taut mooring systems are called semi-taut mooring systems [97].

Tension leg mooring's working principles are as explained for TLPs, and just like a taut mooring system, it relies on the anchor line tensile strength to generate the restoring forces. For heave motions, this elastic mooring line elongation determines the stiffness. For horizontal displacements, i.e. surge and sway motions, the geometric stiffness with constant line length have the largest contribution to the mooring stiffness. The loads experienced by the system's anchors are purely vertical and high, thereby requiring the most expensive anchors [14]. One benefit on the other hand, is that it has the smallest footprint of all systems. Furthermore it has the highest stiffness for pitch and roll rotations, which often is a critical design parameter for FOWTs [14, 118]. The tension leg mooring principle is depicted in figure 2.3 part C.

2.3. Tower and Rotor-Nacelle Assembly

The last main component of a FOWT is the wind turbine itself, which is the component where the electrical energy is being generated. This section presents the components present in the turbine, briefly describes the fundamental working principle of the rotor blades, elaborates on the tower design considerations, and provides some examples of actual turbines.

2.3.1. Components of a wind turbine

The tower supports the RNA, who's rotor blades are a lift driven mechanism to harness the kinetic energy from the wind and turn it into mechanical energy via the torque generated by the mechanism. Although the amount of blades can vary, or even be vertical axis for that matter (e.g. Seatwirl [104]), the technology for capturing wind energy has generally speaking converged to a three-bladed horizontal axis wind turbine [106].

The blades are connected to the turbine's main shaft via the hub, which typically is in front of the nacelle. The nacelle is the housing for all components needed to turn the mechanical energy into electrical energy. On top of the nacelle is an anemometer that measures the wind speed, and a wind vane that measures the mean incoming wind direction. These quantities are used as input for the controller, that operates the yaw gear, the generator torque, and the blade pitch angles. The generated mechanical energy is passed through the main (low speed) shaft towards the electrical generator. Either directly, if it is a direct drive wind turbine, or via a gearbox, if it is a geared wind turbine. This generator

converts the mechanical energy into electrical energy. An overview of the two possible systems is given in figure 2.6.

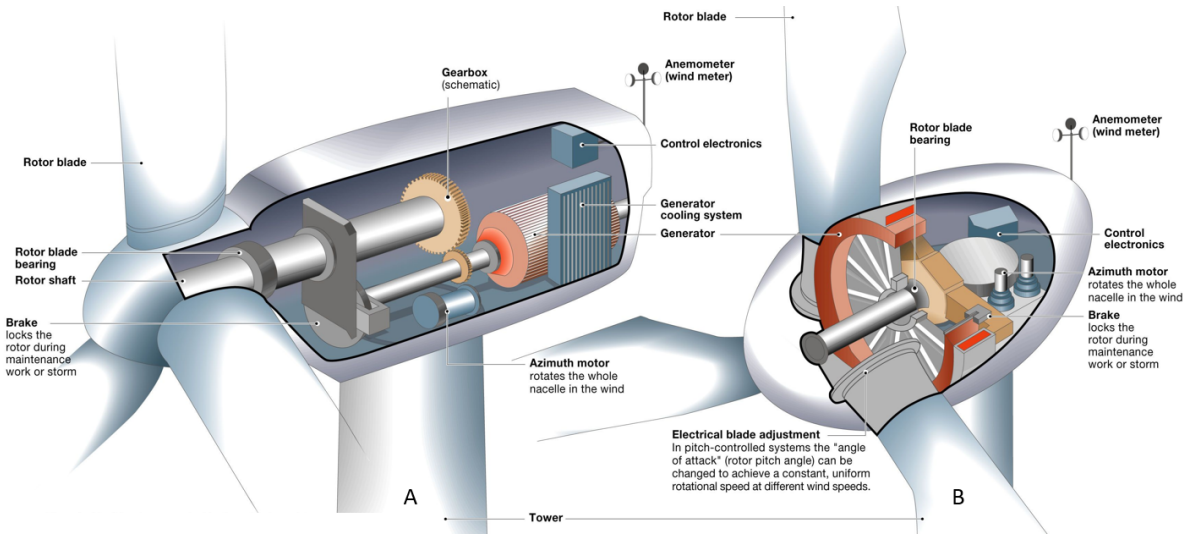

Figure 2.6: Components present in a wind turbine. A: A geared system, B: A direct drive system (retrieved from [118])

2.3.2. Working principle of a rotor blade segment

This section briefly describes the working principle of a rotor blade. The amount of thrust and torque a rotor blade generates are wind speed, rotor speed, and blade angle dependent. To see how these parameters influence the outcomes, a rotor blade cross-section segment is presented in figure 2.7. Note that the blade segment is given as seen from the rotor plane, so that the segment is actually seen sideways from the turbine. Hence, the wind speed experienced by the rotor blade segment due to the rotor speed Ω, is parallel to the rotor plane. The total amount of wind speed experienced by this segment due to the rotor speed is depending on the radial distance r, and the tangential component of the induced flow a'. The induced flow is present because the wind turbine extracts energy from the wind, which affects the wind's velocity and direction. The axial component of the induced flow, a, influences the mean incoming wind speed experienced by the rotor blade segment, U_w. The relative wind speed experienced by the segment, W, is the vector summation of these two contributions.

The angle between the direction of the relative wind speed and the rotor plane, Ψ, is a summation of the blade pitch angle β, and the angle of attack α. The blade pitch angle can be altered by the control mechanism, resulting in a different angle of attack. This in it turn, influences the generated lift and drag of the rotor blade segment. The lift and drag coefficients, c_l and c_d, are angle of attack dependent. The amount of generated lift and drag for the segment can be determined as shown in equations 2.1 and 2.2. The actual values for the lift and drag coefficient can be found in look-up tables for the specific airfoil design [21].

$$dL_i = \frac{1}{2}\rho W^2 c_l(\alpha) c \delta r \tag{2.1}$$

$$dD_r = \frac{1}{2}\rho W^2 c_d(\alpha) c \delta r \tag{2.2}$$

Where c is the chord and δr is the width of the considered segment. The lift is generated perpendicular to the relative wind speed, while the drag is parallel with the relative wind speed.

The lift and drag for the segment can be expressed as a force in the rotor plane and one perpendicular to this, better known as the torque, dQ, and thrust dT. Using the angle of attack, and the local radius of the segment r, the contributions are derived as follows:

$$dQ = r(dL_i \sin\alpha - dD_r \cos\alpha) \tag{2.3}$$

$$dT = dL_i \cos\alpha + dD_r \sin\alpha \tag{2.4}$$

The total torque and thrust generated by the rotor is the integral over the length of the rotor blade times the number of rotor blades.

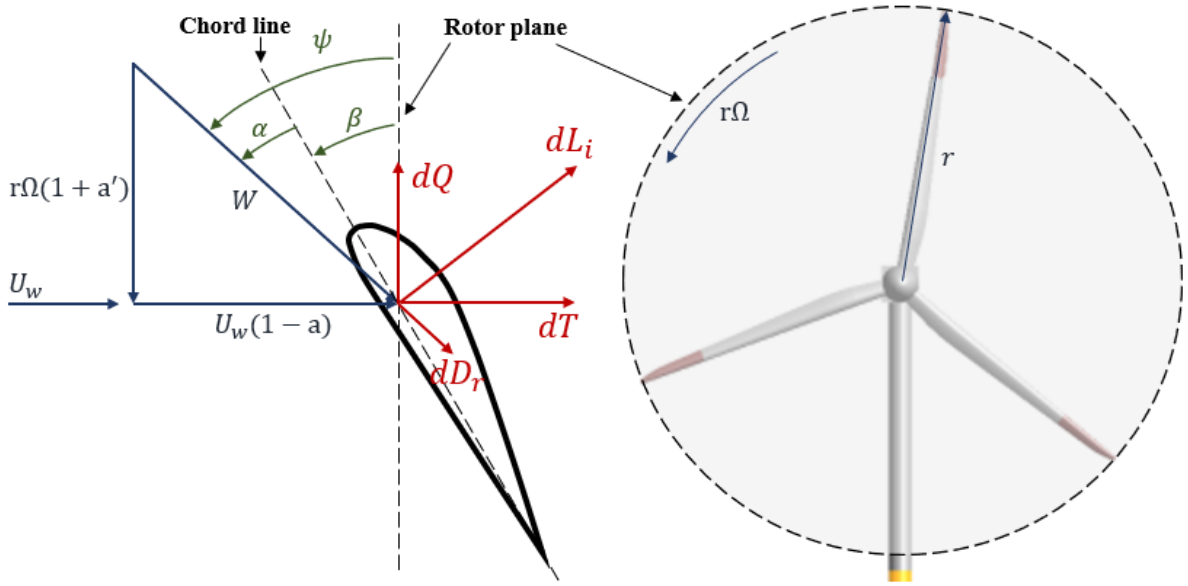

Figure 2.7: Lift, drag, torque, and thrust generating principle for a rotor blade segment

2.3.3. Tower design

The tower makes up a substantial part of the total turbine capital costs, and therefore finding an optimum between its mass and manufacturing expenses is a critical trade-off [82]. The design needs to be strong enough to withstand all the loads induced by the RNA, but at the same time be as light as possible so material usage – and thus costs – is kept at a minimum.

Furthermore, the tower natural frequencies should not be within the range of the turbine rotational speed (1P), nor in the range of the blade passing frequency (3P, for three bladed rotors), to avoid resonance between the tower's first bending mode and the periodic excitation from the rotor. Three frequency ranges can be identified where this is the case: the range below 1P – the so-called soft-soft range, the range between 1P and 3P – the soft-stiff range, and lastly the range above the upper bound of 3P – the stiff-stiff range. The lower limit of the 1P and 3P ranges are determined by the minimum rotational speed of the rotor during operation, the upper limits by the maximum.

Besides the 1P and 3P excitation frequency ranges, offshore wind turbines are also subject to both wind as wave excitation spectra. The wind spectrum is typically described with the Kaimal spectrum [58]. For the wave spectrum, the data collected during the Joint North Sea Wave Observation Project (JONSWAP) is typically used. Note that the JONSWAP spectrum can be modified in such a way that it matches the site specific conditions [48]. Both the 1P and 3P excitation spectra, as the normalized power spectral densities (PSD) for the wind and wave excitation, are shown in figure 2.8 for IEA's 15 MW reference turbine as adapted for floating applications. Note that a safety factor is applied to the blade passing frequencies. This is because of the floating application, the rotor speed can alter due to FOWT motions [3].

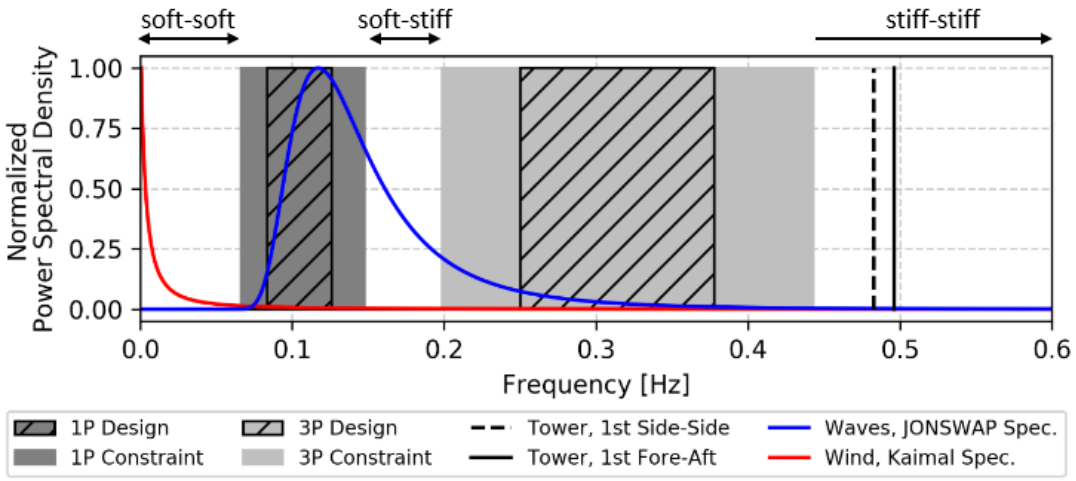

Figure 2.8: Tower natural frequency for the stiffened IEA 15 MW reference turbine relative to the normalized PSD of the excitation frequencies (adapted from [3])

Generally speaking, the ideal range for the tower natural frequencies would be the soft-stiff. In the soft-soft range the structure would be designed too flexible, and moreover, this is a range where environmental excitation may lead to resonance. Furthermore, this range becomes ever smaller for increasing turbine power ratings. At higher rated powers, the turbine blades become larger and the tip speed becomes the limiting factor, resulting in lower rotational speeds. These lower rotational speeds shift the 1P and 3P excitation ranges to lower frequencies. On the other hand, the stiff-stiff range is economically less favorable, as it leads to a more rigid structure which is heavier and more expensive. Therefore the soft-stiff range is often considered as the most favourable [12]. However, for higher rated wind turbines the as mentioned lower becoming rotation speeds make it harder to design a tower with its first fore-aft mode frequency in the soft-stiff range. This can be seen in figure 2.10, where an overview of current designs first fore-aft mode frequencies for different power ratings, both for bottom-fixed as floating applications, is created by Sergiienko et al. [106]. At 20 MW rated power, the soft-stiff range is almost completely overlapped by the wave excitation frequency range. Thus, to avoid resonance with the waves, the high-capacity tower designs for floating applications are typically designed in the stiff-stiff range. For example, the same was the case for IEA's 15 MW reference turbine. The tower used on the VolturnUS-S floater is a stiffer version of the IEA 15 MW reference turbine that is designed for monopiles. This stiffer semi-submersible tower has a mass 47% greater than the bottom-fixed design, which was designed in the soft-stiff range without safety factors included [3, 35].

The stiffer towers for FOWT applications are also due to the additional degrees of freedom that a FOWT has compared to its bottom-fixed counter part. The first structural mode of a FOWT is a rigid-body mode, occurring at lower frequencies. These rigid-body modes create high tower loads due to the weight in the top of the turbine. To deal with these loads the tower needs to be designed stiffer, leading to a higher first fore-aft tower natural frequency [118]. The modes as experienced by the bottom-fixed- an floating turbine are shown in figure 2.9.

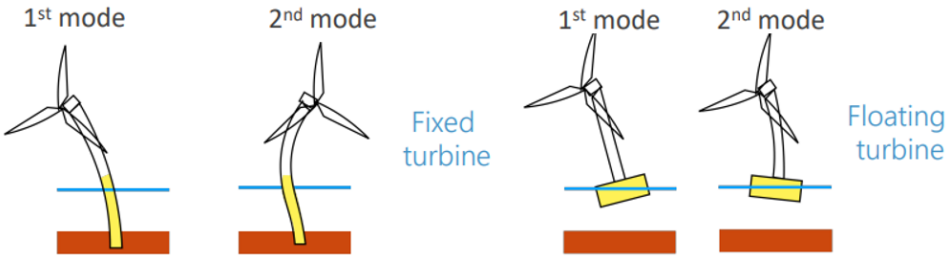

Figure 2.9: First and second structural modes for a bottom-fixed- and floating wind turbine (retrieved from [118])

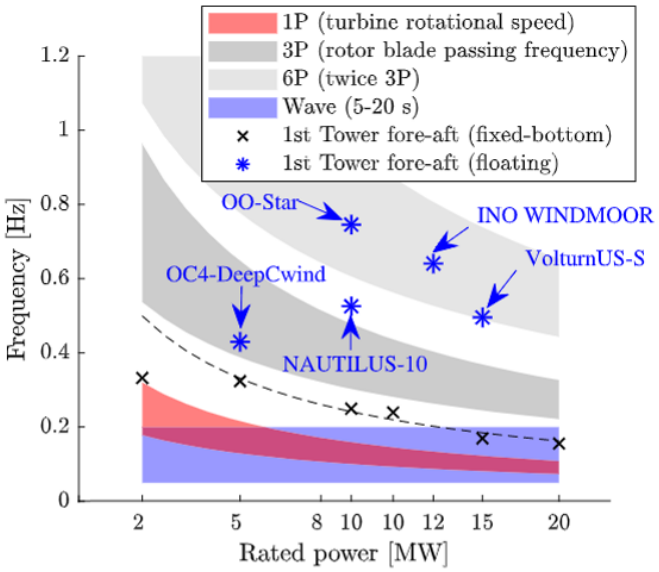

Figure 2.10: Range of excitation frequencies (1P, 3P, 6P and wave) and first fore-aft tower natural frequencies depending on the turbine's rated power (adapted from [106])

2.3.4. Examples

The turbine used for the world's first floating wind turbine had a rated power of just 80 kW, installed on Blue H's prototype floater in 2007 [8]. The first multi-megawatt floating wind turbine prototype was installed in 2009 during the Hywind project, with a rated power of 2.3 MW [27]. The knowledge gained by that prototype led to the world's first floating wind farm, dubbed Hywind Scotland, who became operational in 2017 and consists out of five 6 MW turbines [30]. Recently it is accompanied by the Hywind Tampen wind farm consisting out of eleven 8 MW turbines [31]. Besides the Hywind projects there are other commercial projects ongoing. One such example is the WindFloat project, which started with a 2 MW prototype in 2011 [96], followed by a wind farm consisting out of three 8.4 MW turbines (WindFloat Atlantic) in 2020 and one of five 9.5 MW turbines (WindFloat Kincardine) in 2021. Some of these examples are presented in figure 2.11.

One thing all projects have in common is that they have a trend towards larger turbine sizes over the years [56]. This growth is considered to be the main reason behind the reduction in the levelized cost of energy[1] (LCOE) of wind turbines over the recent years, as increased capacity per unit can induce important savings in transportation, installation, mooring system, operation and maintenance (O&M), and decommissioning [23]. Furthermore, the rotor's diameter size and hub height increase also allow turbines to capture more energy at greater height above ground level, because the wind resource quality becomes better higher above sea level [28].

This trend is also followed by the open sourced wind turbines for floating applications that can be used for demonstrating purposes or academic research. It started with NREL's widely adapted 5 MW turbine [63] in 2009, followed by an open sourced 10 MW turbine designed by Bak et al. [7] in 2013, after which IEA's 15 MW [35] was designed in 2020, and currently the IEA is working on a 22 MW open-sourced turbine [53].

[1] The average price of electricity required to cover the lifecycle cost of a project.

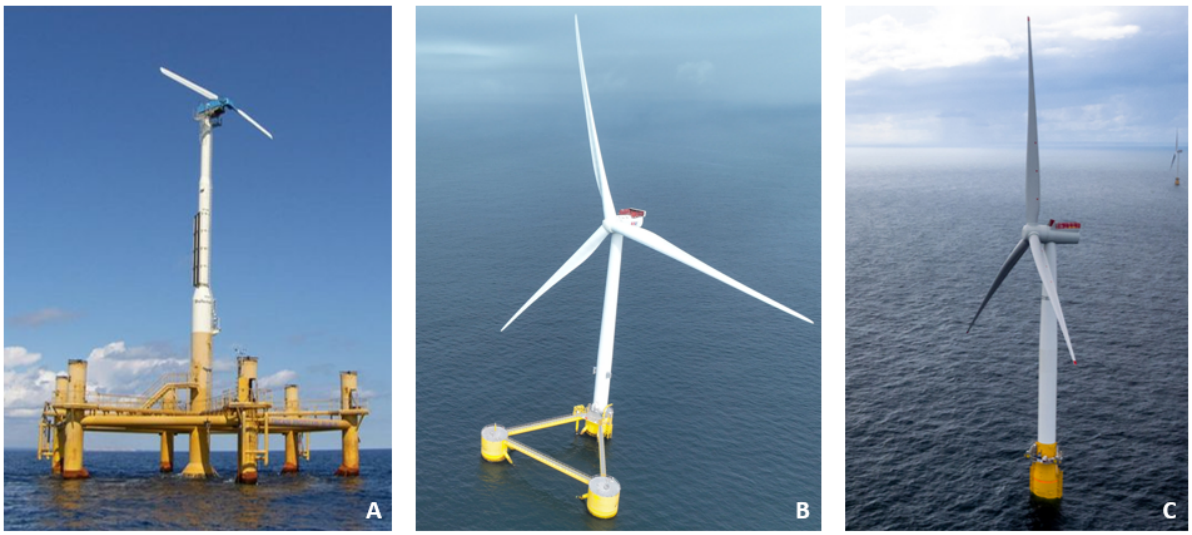

Figure 2.11: Three examples of floating offshore wind turbines. A: Blue H's TLP prototype [8], B: WindFloat's Atlantic semi-submersible [15], C: Hywind's Tampen Spar-type [2]

2.4. Turbine Control

One important topic of a (floating) wind turbine is not yet elaborated on: The controller. It is briefly mentioned as being part of the turbine in section 2.3.1, where it was stated that the control mechanism is situated inside the nacelle and it actuates the yaw gear, the generator torque, and the blade pitch angles. Most wind turbines rely on the control mechanism to alter these quantities to optimize power production, improve power quality and minimize loads on the rotor, generator and tower [23]. By altering the generator torque and the blade pitch angle, and thereby also the rotor speed, the turbine's thrust and torque are altered as shown in section 2.3.2. This in turn influences the dynamic behaviour of the FOWT. How the quantities are actuated can vary depending on the applied control strategy. In some cases, the control action can lead to instabilities. This section starts with explaining these potential instabilities, followed by explaining a widely adapted control strategy, after which more control strategies are discussed. Lastly, an example controller is mentioned.

2.4.1. Floating offshore wind turbine instabilities

In the introduction it is mentioned that the interaction between the FOWT's platform motions and the wind turbine control system may result in serious instabilities, as shown in e.g. [60, 67, 110]. To explain this behaviour, Bachynski et al. [5] and Goupee, Kimball, and Dagher [38] made a Taylor expansion of the thrust. Therefore it is import to know how the thrust of a wind turbine typically changes over wind speed. For every turbine a so-called thrust curve can be generated, wherein the amount of generated thrust at a certain mean incoming wind speed is given over a range of wind speeds. This thrust curve for NREL's 5 MW wind turbine is presented in figure 2.12. The range of wind speeds in this figure is based on the range between the cut-in and cut-out wind speeds. Outside this range, the turbine is not operating and therefore not generating any thrust. At a certain wind speed the rotor speed reaches a point where the generator's maximum power capacity is reached, that wind speed is the so-called rated wind speed. Based on these three wind speeds, two regions within the operational wind speeds can be identified: The below rated region, situated between the cut-in and the rated wind speed, and the above rated region, situated between the rated and the cut-out wind speed. These regions are also shown in figure 2.12.

Now, to demonstrate how the thrust can influence the FOWT's motions, the pitching motion is used as an example. The nacelle dynamics can be represented as a mass-spring-damper system as [112]:

$$(I_{yy} + A_{55})\ddot{\theta} + B_{55}\dot{\theta} + C_{55}\theta = M_y \tag{2.5}$$

where I_{yy} is the moment of inertia of the system around the y-axis, A_{55} is the added mass coefficient and B_{55} the damping coefficient for the pitching motion, C_{55} is the restoring coefficient, θ is the pitch angle, with the dot representing the velocity and double dot the acceleration, and M_y is the excitation

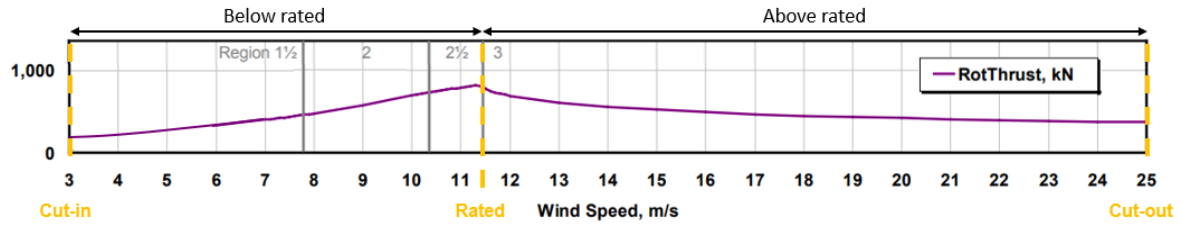

Figure 2.12: Generated thrust at different wind speeds for the NREL 5 MW wind turbine (adapted from [63])

moment around the y-axis. The latter can be estimated as $M_y \approx TH_h$, with T the thrust and H_h being the hub height. Furthermore, the dynamics in surge direction due to the pitching dynamics can be estimated using:

$$\begin{aligned} \theta &= \frac{x}{H_h} \\ \dot{\theta} &= \frac{\dot{x}}{H_h} \\ \ddot{\theta} &= \frac{\ddot{x}}{H_h} \end{aligned} \tag{2.6}$$

To see the influence of the thrust on the dynamics of the system, a Taylor expansion of the thrust is made, keeping only the first derivative and assuming uniform, non-turbulent wind:

$$T = T_0 - \frac{\delta T}{\delta W}\Delta W \tag{2.7}$$

herein T_0 is the initial thrust, and W is the relative wind speed at hub height. The relation between the relative wind speed, and the mean incoming wind speed at hub height, U_w, combined with the system's motion can be estimated as $W = U_w - \dot{x}$. Thereby $\Delta W = W - U_w = -\dot{x}$, so that after substituting the above relations equation 2.5 becomes:

$$(\frac{I_{yy} + A_{55}}{H_h^2})\ddot{x} + (\frac{B_{55}}{H_h^2} + \frac{\delta T}{\delta W})\dot{x} + \frac{C_{55}}{H_h^2}x = T_0 \tag{2.8}$$

Combining this result with a thrust curve as e.g. shown in figure 2.12, it becomes clear that a negative damping can occur in the above rated wind speed region, as $\frac{\delta T}{\delta W}$ is negative in that region, matching with the negative damping phenomenon shown in e.g. [60, 67, 110].

The above mentioned Taylor expansion did not yet include the control action, but also with control included instabilities can occur. This especially happens at lower frequencies of oscillation [112]. The cause of this instability is depicted in figure 2.13, and arises if the controller cannot distinguish the rotor speed fluctuations provoked by FOWT motions (steps C and D in the figure) from those arising from wind fluctuations (steps A and B in the figure), while the controller is fast enough to actuate the blade-pitch during the pitching period [102]. Keep in mind however, that this negative aerodynamic damping does not always lead to an instability, since the hydrodynamic damping in pitch can be significant, especially for semi-submersible and barge platforms [23].

2.4.2. Variable speed variable pitch control system

A potential hazard regarding the stability of a FOWT has been elaborated on in the previous section, but how the actual controller operates is not yet clear. This section describes how a typical controller actuates the system. Large wind turbines mostly adopt a variable speed variable pitch (VSVP) control system [23]. For such a controller it depends on the wind speed region which settings are used to actuate the rotor speed and the blade-pitch angle.

At below rated wind speeds, the power coefficient c_p is kept at its maximum to maximize power production [23]. The power coefficient is defined as:

$$c_p = \frac{P_{out}}{\frac{1}{2}\rho\pi R^2 U_w^3} \tag{2.9}$$

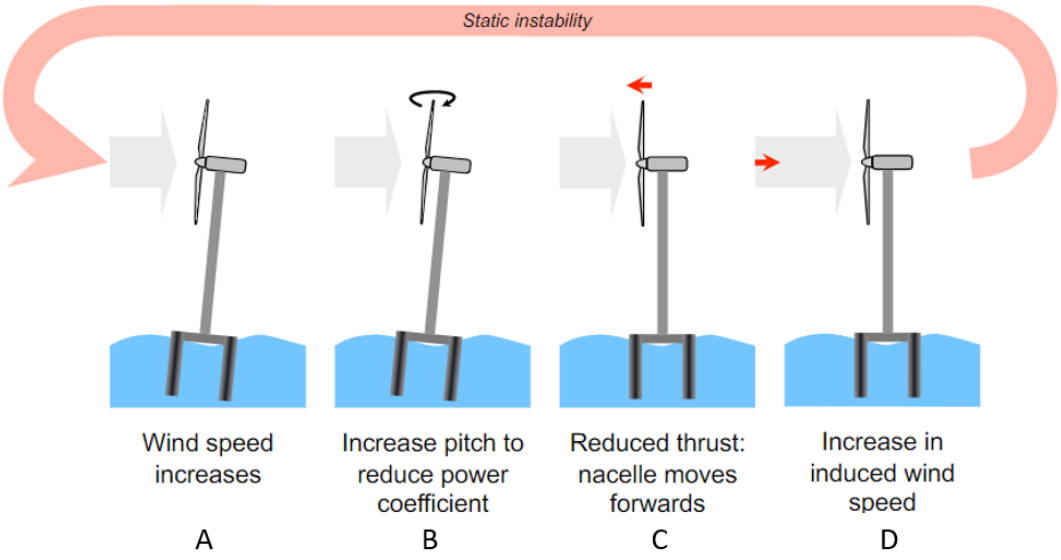

Figure 2.13: Static instability that can lead to negative damping (retrieved from [118])

where P_{out} is the produced power output, ρ is the air density, πR^2 is the rotor swept area, with R the rotor's radius, and U_w is the incident wind speed. The power output, and thus also the power coefficient, is a function of wind speed, rotor speed, and blade pitch angle, as can be seen in section 2.3.2. One way to express how this influences the torque control can be by expressing the generator torque, Q_g as a quadratic function of the low-pass filtered generator speed, ω_g [23]:

$$Q_g = k_{br}\omega_g^2 \tag{2.10}$$

where k_{br} is a function of the maximum achievable power coefficient and the associated tip speed ratio (TSR), λ':

$$k_{br} = \frac{\rho\pi R^5 c_p}{2\lambda'} \tag{2.11}$$

In the above rated wind speed region the controller limits the power capture and rotor loads by actuating the blade pitch angle [74]. The power coefficient drops from its maximum from rated wind speed, to zero at the cut-out wind speed, but due to the faster mean incoming wind speeds, the power output remains maximal throughout this region. This behaviour, including the previous described below rated behaviour, is depicted in figure 2.14

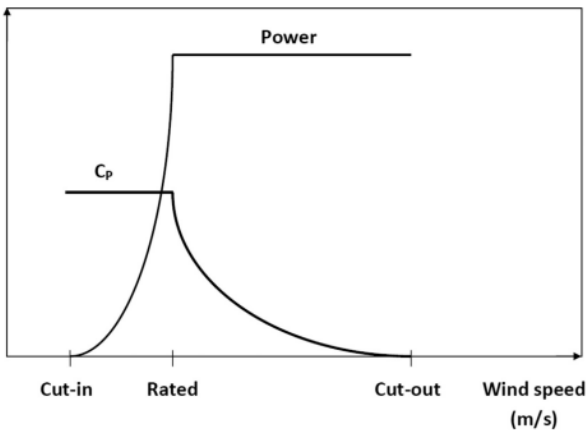

Figure 2.14: Schematic overview of the power output and the power coefficient curves for a VSVP wind turbine (retrieved from [105])

The blade pitch is typically controlled using a proportional-integral (PI) feedback system based on the rotor speed error, $\Delta\Omega$ [74, 23]:

$$\Delta\beta = K_p\Delta\Omega + K_i\int_0^t \Delta\Omega dt \tag{2.12}$$

where K_p is the proportional gain and K_i the integral gain. In the above rated wind speed region, one can choose to keep the generator torque constant, or to maintain the maximum power output. For the latter the generator torque can be varied, using the rotor speed [23]:

$$Q_g = \frac{P_r}{N_g\Omega} \approx Q_{g,r} - \frac{P_r}{N_g\Omega_r^2}\Delta\Omega \tag{2.13}$$

where subscript r denotes that it is the parameter's value at rated power, and N_g is the drive train gear ratio.

For example, the rotor speed and blade pitch angles of NREL's 5 MW wind turbine are presented in figure 2.15, clearly showing how the blade pitch angle is being actuated from rated wind speed and up, while that is the region where the rotor speed is kept constant.

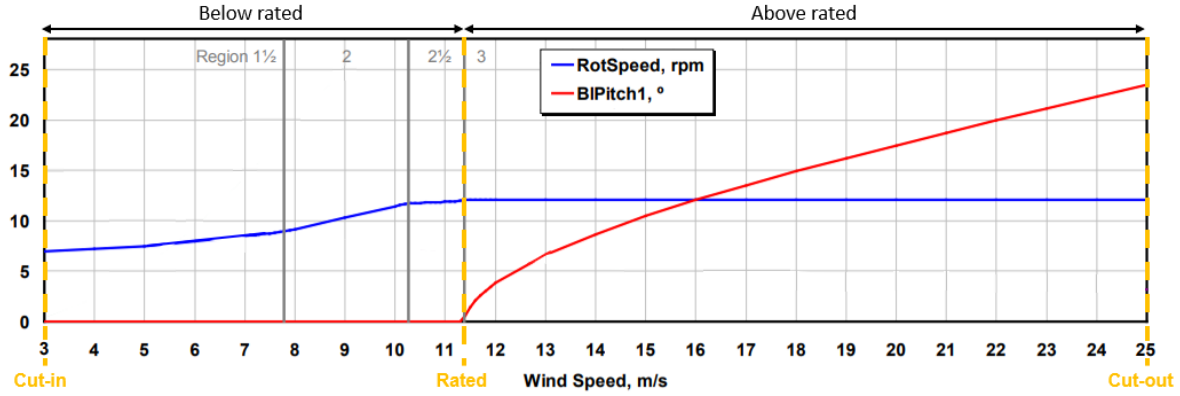

Figure 2.15: VSVP controller operating regions for the NREL 5 MW wind turbine (adapted from [63])

2.4.3. More control strategies

Besides the described VSVP control strategy, other (additional) control strategies can be applied, amongst others to avoid the stability problem. Some of these options are described in this section.

Detuned controller

The simplest option is to detune the controller. By reducing the gains, the controller reacts more passively to apparent wind fluctuations so that the controller natural frequency becomes lower than the FOWT pitch natural frequency [67], and so essentially filtering out the controller's response to the FOWT's pitch movement. A drawback to this method is that a detuned controller leads to larger rotor speed excursions, which negatively impact power quality and provokes larger variability of the rotor loads. Furthermore, the detuned controller leads to marginal aerodynamic damping of the pitch movement. Although the instability is avoided, the pitch motions tend to persist for longer than they would in the presence of more aerodynamic damping [23].

Nacelle velocity feedback controller

To avoid these longer pitch motions from happening, control methods can be applied that keep the higher aerodynamic damping but try to avoid the instability from happening. One option to do so is by adapting a motion compensation strategy, so that the FOWT's motions are used as an input of the controller, meaning that the controller is anticipating on the FOWT's motions. This can be via direct feedback of the measured nacelle velocity, which is then taken in consideration in the blade-pitch control action [61]. This method is referred to as the nacelle velocity feedback method. Since in this method the FOWT's motions are taken into account in the control action, the potential instabilities due to control action as a result of these motions as described in section 2.4.1 are avoided.

Although the method is proposed because the turbine has additional motions due to being placed on top of a floater that can cause instabilities, it also means that the nacelle velocity also contains wave-frequency components as a result of the wave-induced forces. Although at floater level the pitch motions in the wave-frequency range are generally small, the resulting tangential velocity at the nacelle height may become a significant contribution, inducing non-negligible wave-frequent components to the blade pitch command [23]. As described in section 2.4.1, the potential instabilities occur in the low frequency range, as that is the range where the controller is fast enough to actuate based on the FOWT's motions in addition to the environmental conditions. Since the wave-induced motions are at higher frequencies, it is desirable to low-pass filter the nacelle velocity before using it in the control strategy. This way, the controller uses the FOWT's motions in the control action in the frequency range where it is fast enough to actuate as a result of these motions, whereas it does not include them in the higher frequency range where the controller is not able to actuate based on these motions. However, low-pass filtering can induce a significant phase lag to the filtered signal. This can affect the stability of the controller, and should thus be included in the stability analysis. If the nacelle velocity feedback method is used in combination with low-pass filtering of the measured nacelle velocity, it could prove difficult to find stable gains [23].

Nacelle velocity feedforward controller

Another method is to reject the controller from actuating the rotor speed based on the FOWT's motions, before the rotor speed error is fed into the controller [23]. This method is the so-called nacelle velocity feedforward method. There are multiple ways of implementing this idea. For example, it can be done via an observer that separates the rotor speed components provoked by fluctuations in wind speed from those arising from the FOWT motions, and then feeding only the former into the controller [23, 109]. Another, simpler way of doing it is by modifying the reference rotor speed that goes into the controller based on the measured nacelle velocity [23, 66]. By using the nacelle velocity for calculating the rotor speed reference, the controller is informed whether the turbine in pitching downwind or upwind, thus avoiding the rotor speed variations caused by the FOWT motions being fed into the controller [23].

This method uses the nacelle velocity as an input to determine the control action, just as was the case for the nacelle velocity feedback method. Therefore, the same reasoning regarding the low-pass filtering also applies for this method.

Peak shaving controller

Besides the motion compensation strategies there is another control strategy that can be adapted, based on the thrust curve of the turbine. As shown in equation 2.8, the magnitude of the aerodynamic damping is depending on the derivative of the thrust with respect to wind speeds, which peaks at rated wind speed, as can be seen for NREL's 5 MW reference turbine in figure 2.12. Note also that this derivative is the steepest around rated thrust, thus the negative damping in the above rated region is highest around the turbine's rated power. To avoid the wind speed range with the most negative damping effect, thrust peak shaving can be applied. Instead of maximising power production until rated wind speed, the rotor thrust is limited at a range near rated wind speed [33]. It does so by already starting to actuate the blade pitch angle before the rated wind speed. Fischer and Shan [33] represented this behaviour for both a situation with and without thrust peak shaving active, shown in figure 2.16. The method can also be desirable if the objective would be to lower the maximum tower base loads, since the maximum thrust – and thus the bending moment due to the thrust force – is kept lower.

The draw back of this method is that as a consequence of reducing the power coefficient already before rated wind speed, the power output of the turbine is also kept lower at the wind speeds where peak shaving is applied. Therefore, when one design a peak shaver a trade-off between the loads and the energy yield must be made [33].

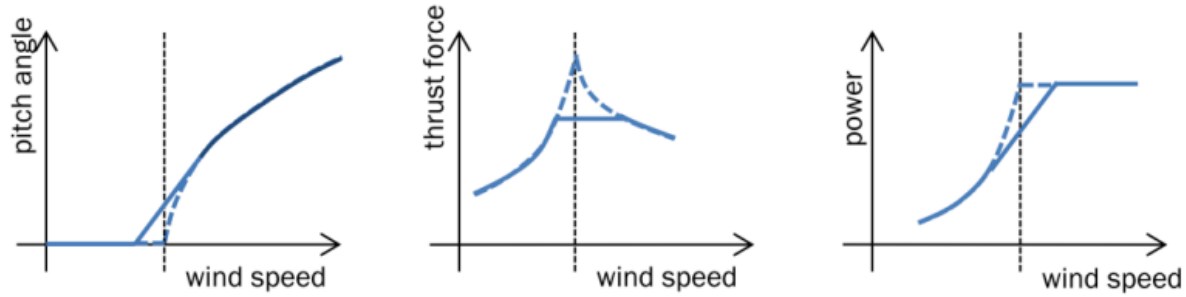

Figure 2.16: Steady operating points with peak shaving (solid line), and without (dashed line). The vertical black dashed line represents the rated wind speed. (retrieved from [33])

More control strategies

All the control strategies mentioned so far assume collective blade pitch control, i.e. all the blades have the same blade-pitch angle at all times. Another control method would be to command specific increments to each blade on top of the collective command. This method is called individual or cyclic pitch control. One of the main reasons to adopt this is to reduce excitation at blade passing frequencies. The commands may be determined in order to compensate for the different wind velocity seen by each blade, caused by wind shear, tower shadow effects, and turbulence [23]. However, effective reduction of blade passing related loads with this method is only possible if the asymmetric loading on the rotor can be measured with accuracy [11]. So even though the method can decrease structural costs and increase energy production, it has a negative impact on the reliability of the turbines because many sensors must be used [102].

Another alternative are smarter artificial intelligence (AI) control methods. Work has been done on e.g. neural network based pitch controllers [22, 64], which require a large amount of real-time data from the system [102]. Since these methods use black-box approaches they are not of further interest for this thesis, as they do not contribute to a better understanding of the physics involved.

2.4.4. Example controllers

The description of the control strategies mentioned above are their fundamental working principles. Actual controllers are often more extensive, but their precise working principles and gain values are kept as a company secret. For academic and open-sourced research purposes NREL's Reference OpenSource Controller (ROSCO) tool-set is often used [1, 89]. It is specifically designed to ease controller implementation for the wind turbine researcher.

2.5. Operating conditions of a floating offshore wind turbine

Now all the components present in a FOWT are described, one can look at the environmental conditions a FOWT is operating in to better understand what each component has to withstand during its operational lifespan. The total amount of physical phenomena that could potentially influence the response of a FOWT is astounding. Jonkman [62] created an overview the various environmental loads a FOWT might encounter during its time in operation, as illustrated in figure 2.17. Some of these environmental conditions are only present during extreme (weather) events, such as the mentioned earthquake or ship impact. For this thesis, only the contributions influencing the response during the nominal operating conditions are considered, i.e. the hydrodynamic and aerodynamic excitation, as well as the components influencing the dynamic response, i.e. the mooring system and the control system.

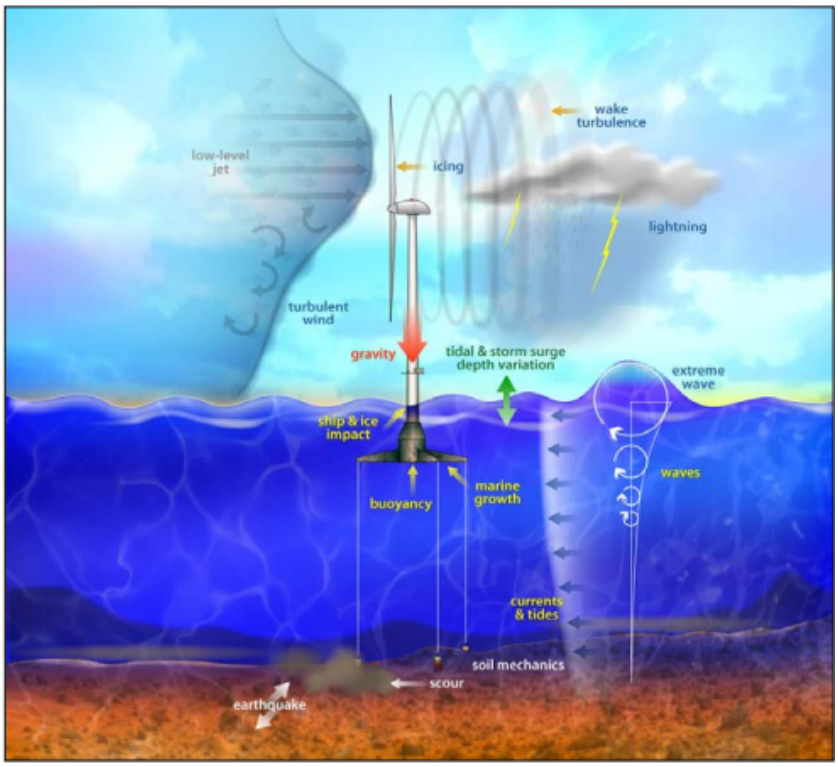

Figure 2.17: Potential environmental phenomena influencing the response of a floating offshore wind turbine (retrieved from [62])

2.5.1. Hydrodynamic loads

The hydrodynamic loads are induced by the waves and currents acting on the system. The waves generate loads that include wave frequency dynamic loads and wave drift forces. Ocean waves are irregular and random in shape, height, length and speed of propagation. In order to model ocean waves, a linear random wave is modeled as the superposition of many small linear wave components with different amplitudes and frequencies. Figure 2.18 shows an example of such a superposition. To represent the random waves, the wave phases are random with respect to each other. Using this the waves can also be represented by an energy spectrum that gives the distribution of wave energy among different wave frequencies and heights on the sea surface. The most common used spectra to do so are the Pierson-Moskowitz and JONSWAP spectrum. The Pierson-Moskowitz represent a fully developed sea state, whereas the JONSWAP is a fetch limited version of the Pierson-Moskowitz spectrum [72]. The JONSWAP can be modified in such a way that it matches the site specific conditions [48].

Currents are typically defined by direction and a velocity profile that varies of the water depth. For most applications the current velocity can be considered as a steady flow field where the velocity magnitude and direction are a function of water depth. The currents generate drag and lift forces on the submerged structures. The waves and current can interact, with as a result that the mean wave drift forces are affected by the current [72].

The hydrodynamics are subject to nonlinearities. If waves become too steep they can break. If this happens at the location of the structure it results in a sudden and intense impact, resulting in a complex response. Furthermore difference frequencies arise due to nonlinear interaction between waves of difference frequencies. At higher velocities, the floater's viscous damping effects may become nonlinear, as they might become proportional with to the square of the velocity. Also, the hydrodynamic loads interact with the restoring forces. If the force is so large that the body significantly moves, the underwater body's shape may alter in such a way that a nonlinear relationship occurs between the displacement and the restoring force. Also, the mooring system's stiffness is not linear over its offset, introducing nonlinearities depending on the structure's motion as well.

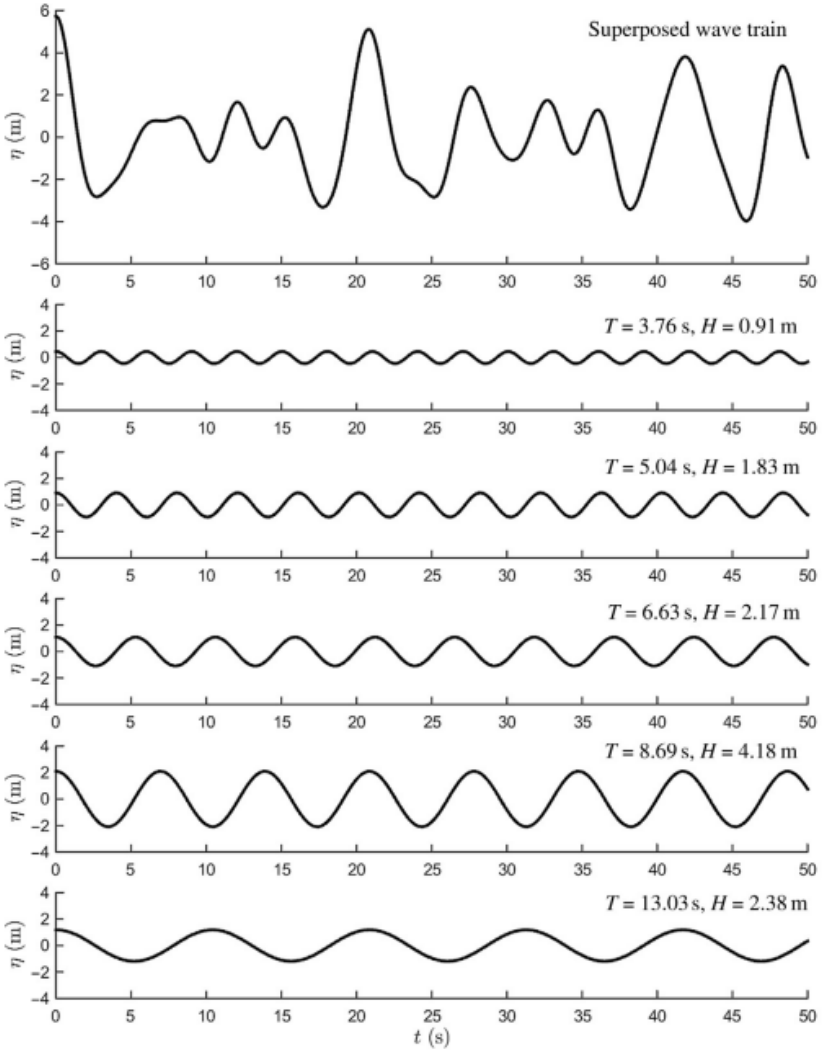

Figure 2.18: The superposition of multiple regular waves to generate a irregular wave (retrieved from [72])

2.5.2. Aerodynamic loads

The aerodynamic loads are induced by the wind. The wind characteristics can be described by both long term and short term spectra. For the long term it can be captured in a Weibull distribution, wherein wind speed variations during the year are characterised. The short term, often referred to as turbulence, describes wind speed fluctuations on a relatively fast timescale. Turbulence is a complex process due to the many parameters influencing the characteristics, such as the temperature, pressure, density, and humidity as well as the motion of the air itself in three dimensions. Thereby it cannot be represented by terms of deterministic equations. Therefore it is generally more useful to develop descriptions of turbulence in terms of its statistical properties. The spectrum of turbulence describes the content of short term wind speed variations, examples are the Kaimal and von Karman spectra.The Kaimal spectrum gives a good fit to empirical observations of atmospheric turbulence [58].

As shown in section 2.3.2, the aerodynamic forces rely on the angle of attack. The relationship between the lift- and drag forces and the angle of attack is nonlinear, especially as the angle approaches stall conditions. Therefore, wind gusts have a nonlinear response by altering the angle of attack. For the same reason, the control action can induce nonlinear behaviour. Furthermore, the turbine's wake introduces complex and nonlinear behaviours, especially if a turbine is operating in the wake of another. Therefore, research is being done in wake turbulence optimization for wind farms [17].

2.5.3. Hydrodynamic and aerodynamic interaction

As stated in chapter 1, FOWTs have a remarkable multidisciplinary character. The total system's behaviour cannot be solved by simply analyzing every sub-component as mentioned in the previous sections individually. Forces on one component will induce a response and excitation that subsequently impact another component, and therefore the overall system behaviour. This relationship make the system's response even more complex.

One example of this phenomenon is that a FOWT can become operating in its own wake field due to a surging or pitching motion. This happens when the turbine moves backwards faster that the wake field development, as can be seen in figure 2.19. This induces nonlinear loads on the turbine's blades that a bottom-fixed turbine would not encounter.

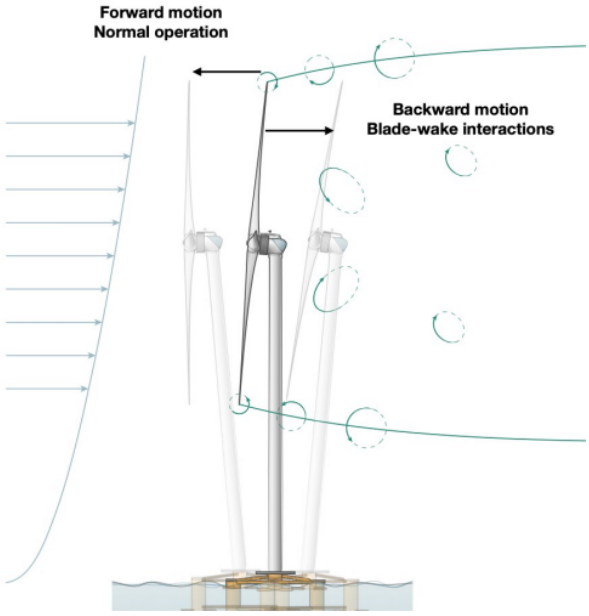

Figure 2.19: Pitch motion resulting in a turbine operating in its own wake field (Retrieved from [118])

Not all turbine-floater interactions have to be this extreme. Some of the FOWTs nonlinearities come from the interaction between the aerodynamic forces and the hydrodynamic forces. The interaction between the rotor-induced flow field and the ocean waves introduces nonlinearities in the system response, affecting both the aerodynamic performance of the turbine and the platform motion [17].

2.6. Most Favourable Design

In the previous sections all the components of a FOWT and the environmental loads a FOWT is subjected to are explained. For each design choice, pros and cons are mentioned. As mentioned, already more than 30 FOWT concepts have been proposed. None of them has proven itself to be outperforming the others, and therefore the market did not yet converge to one design type [68]. That is to say, hitherto there is no such thing as a proven most favourable design.

Goupee et al. [39] examined the behaviour of three floating designs – a Spar, TLP and semi-submersible – for a range of wave and wind loads, and concluded that preference generally depends on onsite conditions. However, according to Leimeister, Kolios, and Collu [68] it could prove beneficial for a more competitive LCOE if the industry would focus on one concept to optimize.

Besides the technical pros and cons, logistical conditions during the life-cycle of a FOWT have been shown to be a limiting factor [68]. As mentioned, the semi-submersible has the advantage of the shallow water depths required combined with self-stabilizing properties, which enables assembly in controlled environments and low vessel cost necessity. A basic tug boat could be sufficient for wet-towing to the desired location [19]. Although the latter also holds for Spar designs, these require evermore draught for increasing turbine sizes. For the O&M McMorland et al. [80] performed a literature study, wherein they state that the TLP will be the most effective regarding this topic, both in terms of cost and ease of maintenance, despite the fact that no TLPs have been installed at commercial level.

Based on these observations, the semi-submersible and TLP are therefore interesting design ideas to be investigated further, each having their own advantages and disadvantages. However, in the offshore wind market report by Musial et al. [83], it is pointed out that currently about 80% of the commercial floating wind projects installed and in the pipeline plan to use a semi-submersible floater, as shown in figure 2.20. Therefore it is chosen that the main focus of this thesis will be on semi-submersibles.

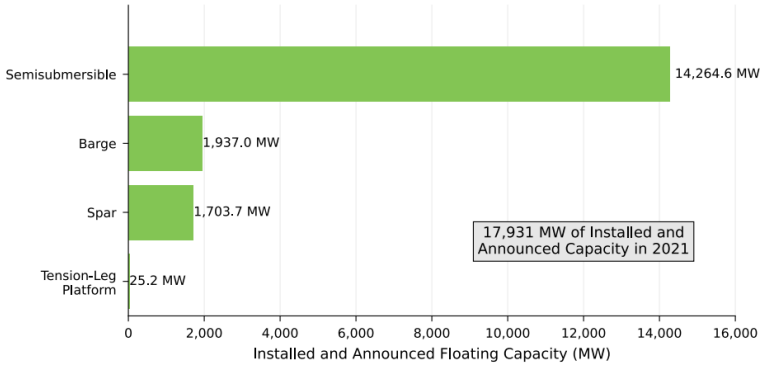

Figure 2.20: Global FOWT substructure market share, installed and announced cumulative (retrieved from [83])

Within the design category semi-submersibles, there are still different design ideas. For example, two different open-sourced semi-submersible floater designs for the open-sourced 5 MW turbine are visualized by de Souza and Bachynski [24], as shown in figure 2.21. The OC4semi, part A in figure 2.21, has the turbine positioned on a central column, which is connected to three side columns using thin braces. Each of the side columns have a so-called heave plate at the bottom, which are segments with a larger diameter with the purpose of increasing the added mass and damping in heave direction. The other design, the CSC5MW, part B in figure 2.21, does not have these heave plates. In stead, the side columns are connected to the central column using pontoons. These pontoons contribute to the overall buoyancy, and to the added mass and damping in heave direction. There are also semi-submersible design with the turbine placed on one of the outer columns, such as WindFloat's Atlantic semi-submersible, shown in figure 2.11 part B.

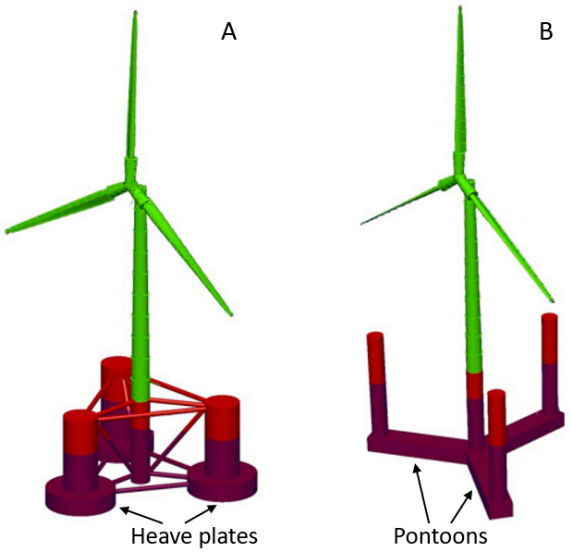

Figure 2.21: Two examples of semi-submersible floater designs. A: The OC4semi design, B: The CSC5MW design (retrieved from [24])

The performance of all these different designs have to be assessed before they will be manufactured on full scale, not just the floater design, but the whole FOWT design in its entirety. There are multiple options to do this assessment. Chapter 3 discusses the available options there are to do this.

3

Modelling techniques

The previous chapters emphasized that the FOWT technology is still in their preliminary stage of technology readiness, and that there are many different FOWT concepts being proposed. These concepts have to be assessed to evaluate how the system operates in different environmental conditions. This chapter discusses the different possible options that can be used to investigate the performance of such a FOWT concept. By looking at the pros and cons of these options, while keeping the current stage of technology readiness in mind, the numerical frequency-domain method (FD) is chosen as the preferred method to improve on. The method is described in more detail compared to the other options, and the state-of-the-art of this method is elaborated on. Finally, the potential improvements that can be made in addition to the state-of-the-art are stated.

3.1. Optional methods

A variety of physical and numerical modelling approaches can be used to predict the dynamic response of a FOWT, although it is challenging due to the strong coupling between the aerodynamics of the turbine and the hydrodynamics of the floating structure. This makes the dynamic response of the whole system complex. Not all the modelling techniques take all the (nonlinear) loads as mentioned section 2.5 into consideration. This section briefly describes the possible modelling techniques, what their assumptions or limitations are, and how these influence the usability of the results of each technique. Otter et al. [92] gathered the state-of-the-art modelling techniques that will be presented in this section. Most information is gathered from this source, unless explicitly stated differently. At the end of the section, it is explained why the frequency-domain method is chosen as the preferred method to improve on.

3.1.1. Experimental testing

Experimental testing is considered as an essential step in the design process by most FOWT designers. The experimental results obtained in laboratory basins are used to validate numerical results. To ensure that the model behaviour is approaching full scale behaviour, the appropriate scaling law has to be used. Geometric, kinematic and dynamic similarity between model scale and full scale is required for the experimental validation of numerical models.

For geometrical similarity this means that all linear dimensions must have the same scale ratio λ, which is defined as:

$$\lambda = \frac{L_F}{L_M} \tag{3.1}$$

where L is the length and the subscripts F and M indicate full scale and model scale, respectively.

The ratio between inertial forces and gravitational forces is defined by the Froude number Fr:

$$\mathrm{Fr} = \frac{U^2}{gL} \tag{3.2}$$

where U is the velocity and g is the gravitational acceleration. The dynamic similarity requirement between model scale and full scale is then defined as:

$$\frac{U_M}{\sqrt{gL_M}} = \frac{U_F}{\sqrt{gL_F}} \tag{3.3}$$

The ratio between inertial forces and viscous induced forces is defined by the Reynolds number Re:

$$\text{Re} = \frac{UL}{\nu} \tag{3.4}$$

where ν is the kinematic viscosity. The dynamic similarity requirement between model scale and full scale then becomes:

$$\frac{U_M L_M}{\nu} = \frac{U_F L_F}{\nu} \tag{3.5}$$

In practice, this means that the only scale ratio that meets both dynamic similarity requirements derived using Froude's and Reynold's scaling would be a one-to-one model, as the kinematic viscosity of the air will remain the same for full-scale and model-scale. This means that a choice between applying Froude scaling or Reynolds scaling law for the model tests has to be made. The choice of which scaling law is followed is depending on the experimental modelling technique used. The following subsection will explain which methods there are, and whether Froude or Reynold caling is used for that method.

Full physical testing

Model testing of a FOWT can be done in multiple manners. If the system as a whole, i.e. the turbine and the floater together, is scaled down and tested at the same time, it is referred to as full physical testing. For this type of testing the Froude scale is used, because in the case of water flow with a free surface, gravity is the dominant factor, overshadowing the impact of other factors like viscosity, surface tension, roughness, etc., which are typically minor and thus can be disregarded. In such instances, Froude's law is the most relevant [20]. Therefore, this law is used for marine applications.

One drawback for a Froude scaled rotor with geometric similitude is that the Reynolds number will be lower at model scale compared to full scale. Consequently, the lift coefficient of the model scale rotor will be lower and the drag coefficient will be higher compared to the full scale rotor. As a result, the aerodynamic performance of the model scale turbine will be affected, causing a misrepresentation of wind load induced motion response of the FOWT. From equations 3.1 to 3.5 it becomes clear that either increasing the velocity of the applied wind or adjusting the turbine rotor geometry at model scale would compensate for the lower Reynolds number. With the increased wind speed method the tip speed ratio (TSR) cannot be maintained, with the result that rotor thrust and torque cannot be modelled correctly, and as a consequence aerodynamic damping may not be captured correctly. Furthermore, a different TSR also means that resonance due to blade passing frequencies is not simulated correctly. To maintain the same performance and TSR as the full-scale counter part, the second option of geometrically modified aerofoils can be applied. The blades are designed so that they imitate the aerodynamic loads even though they are operating in a Froude scaled wind environment. As a result, they capture the damping effect and blade pitch control effects while maintaining TSR, thus also maintaining the same blade passing frequencies. For these reasons adapting geometrically modified aerofoils is the preferred method for full physical testing. In practice, correct thrust modelling is favoured over torque, as the thrust has a much bigger impact on the motion of the FOWT. Hence, achieving the desired scaled thrust is the primary objective in rotor scaling. When the scaled rotor can generate the required thrust for specified wind velocities and rotor speed, according to Froude laws, the rotor is considered to maintain its performance at the model scale [42]. An example of model turbine with modified airfoils is presented in figure 3.1 part A, which could also be placed on top of a floater.

Since this method physically has all components present, it is the least sensitive to numerical errors and uncertainties, although it remains sensitive to scaling effects. However, it is also the most expensive option because the facility needs to simulate both sea and wind states, and the model's rotor blades, engine, actuation system and sensors, must specifically be designed to match the aerodynamic loads [42].

Hybrid testing

Hybrid testing is the alternative to full physical testing. Either the wave or the wind loads are still physically generated, but the other loads are replaced by a numerical actuated substructure. Such methods are referred to as software-in-the-loop (SIL) methods. If the waves are still generated physically, the Froude/Reynolds mismatch is solved by calculating the aerodynamic loads at full scale and apply them to the physical model at Froude scale via one or several mechanical actuators. An axemaple of this method using winches to simulate the aerodynamic forces is given in figure 3.1 part C. The other approach is to use SIL to numerically represent the hydrodynamics of the system. This approach enables validation of aerodynamic forces in an environment with superior wind quality compared to wind generation over a wave basin, while allowing for examination of these aerodynamic forces under motion typical of a FOWT. This is particularly useful when studying the evolution of turbine wake of FOWTs, and how it may affect downwind turbines and farm layouts. An example of this approach is given in figure 3.1 part B.

To study the full performance of the FOWT the best results are obtained by performing both SIL methods. A benefit of using both hybrid approaches is that they each investigate the same problem from two different perspectives, which can be used to optimize the design and study of many components of FOWTs. For instance, diverse control strategy or rotor-blade aerodynamics can be tested in the basin for a given physical floating system, while various hydrodynamic designs and mooring system configurations can be tested in the wind tunnel for a given physical wind turbine [42].

As the methods are still relatively new, the full physical testing method can be used to validate hybrid methods, and vice versa. Such a validation increases confidence in the SIL applied in hybrid testing [42]. For all model test however, it still holds that they are relatively costly compared to fully numerical models. Models tests are therefore only used towards the end of the design progress.

Figure 3.1: Examples of model testing. A: A turbine with modified airfoils on a fixed support [42], B: A turbine on top of a hexapod to simulate hydrodynamic forces [44], C: A floater with winches to simulate aerodynamic forces [42]

3.1.2. Numerical modelling

Besides physical modelling, one can also simulate the system using software only: Numerical modelling. It is a cost efficient way of simulating and analysing different design iterations, before committing to the investment of constructing a prototype or product. A broad range of numerical techniques and software packages are available for the initial stages of the design process of a FOWT. The choice of numerical modelling method will generally be a trade-off between on one hand the accuracy and/or fidelity, and on the other hand the computational efficiency with respect to the phenomena of interest and their acceptable performance levels. Overall, numerical models may be classified into three relative levels: low-, mid- and high-fidelity, where increasing fidelity typically leads to a higher demand on computational resources, reducing computational efficiency. The choice of fidelity level is a function of the objective of the simulation and the level at which such analysis and its accuracy is required. Each of the options will be briefly discussed in the next subsections.

Low-fidelity

Usually, low-fidelity models are used during the initial stage of FOWT design for sizing analysis and first optimization steps. This could be for the testing of new designs, but also for e.g. existing floater designs that need to be sized in order to fit a larger turbine as was examined by Sergiienko et al. [106], or for a design that is optimized for certain site specific conditions and needs sizing to keep the responses in different environmental conditions the same at hub height. In the early design stages, low-fidelity models can be used to simulate linear dynamics of the FOWT. Frequency-domain (FD) tools are often used for floating structures. Typically they work by constructing a linear, frequency-dependent representation of the system and then solving for the harmonic, steady-state system response at each excitation frequency. This linearization makes that it calculates the system's coupled response orders of magnitude faster than higher order fidelity tools, and therefore it is the fastest method [71], but at the same time also the least accurate. A linear representation of a nonlinear system is valid only for small deviations from the original position at which the system is linearized [62]. Therefore, caution is required when using the FD approach for scenarios where nonlinearity might be significant. Furthermore, the method is not capable of predicting loads due to e.g. slamming events. For these time-specific occurrences a fidelity tool of a higher order is needed.

Even though the FD approach is adapted in the early design stages of FOWT design and there are still many FOWT concepts being proposed, so far only one coupled open-sourced method is available for FOWT applications: RAFT – acronym for Response Amplitudes of Floating Turbines [88]. Other software exist for floating applications (e.g. DIFFRAC, WAMIT), but these miss the capability of linearizing the aerodynamics at the same time. As seen in section 2.5, the aerodynamics and hydrodynamics are strongly coupled, and therefore the aerodynamics should be included for more accurate results. So although the maritime industry is experienced in using FD methods to solve the hydrodynamics, coupling this knowledge with frequency dependent aerodynamics is new. Currently this coupling issue is most often omitted by using separate time-domain preprocessing of the turbine properties. The problem with this solution is that it has to be redone every time the turbine design is altered [25, 46].

Mid-fidelity

One fidelity level higher than the low-fidelity models are the mid-fidelity models. These are typically used after the initial design stage for global loads analysis of FOWTs during operational and extreme conditions, both in linear and nonlinear loads. It must be mentioned that time-domain simulations can also be used to solve the coupled dynamics of the system for the initial design stage, but they can have extensive calculation time. For instance, when evaluating all the design load cases (DLCs) of a FOWT it can take up to six to eight weeks of computational time for a single design iteration [79], making it too expensive to use for first design iterations.

Examples for FOWT application are OpenFAST and aNySIM, but many more options exist, an overview is presented in the paper by Otter et al. [92]. The programs are so-called time-domain (TD) programs, and simplify the Navier-Stokes equations so that the response can be calculated faster by the use of potential theory. Potential flow assumes the flow to be irrotational and inviscid. As a result, the tools are less accurate compared to Computational Fluid Dynamics (CFD) tools, but TD tools can still include nonlinear and transient effects while being computationally faster. Most TD programs solve the equations of motion at every time step using hydrodynamic coefficients which are pre-processed with a FD potential flow solver. Limitations of potential flow methods are that viscous effects are being ignored, and that oscillation amplitudes are assumed to be small compared to the cross-sectional area of the floating body. However, since TD tools are capable of taking nonlinear and transient effects into consideration, and therefore model the responses more accurately compared to FD tools, TD tools can be used as validation for FD tools.

High-fidelity

High-fidelity models are typically used during the final design stages for detailed investigations, especially to accurately obtain stresses on the structure. This kind of software is, for example, used for detailed investigations of local flow phenomena and stress hot spots in the structure, as well as for increased accuracy, especially for extreme conditions. One example of a high-fidelity software designed for FOWT applications is QBlade. Furthermore a number of both open-sourced (e.g. OpenFOAM) and commercial software (e.g. Ansys, REFRESCO) packages available for general engineering purposes that can also be used for FOWT design. These programs are so-called CFD (Computational Fluid

Dynamics) programs. Their method is based on solving the Navier-Stokes equations. Depending on the goals of the analysis CFD can be coupled to lower-fidelity tools, e.g. if interested in the wake of a turbine, the turbine itself is analysed using CFD, while the hydrodynamics of the floater can be simulated using lower-fidelity programs to reduce computational efforts. Note that the CFD software is only capable of determining the forces due to flows around a surface, so internal loads are not taken into consideration. If one is interested in these loads, CFD software can be coupled to e.g. Fine Element Method (FEM) software, that executes the structural analysis. These methods are capable of taking all the nonlinearities as mentioned in section 2.5 into consideration, but that comes for relatively very high computational expenses.

Preferred method to improve on

In chapter 1 it was stated that the FOWT industry is still in their preliminary stage of technology readiness, and in section 2.6 the conclusion was drawn that the there is no such thing as a proven most favourable design, with new concepts still being proposed. With the knowledge gained in section 3.1.2, one can imagine that therefore the frequency-domain method is the chosen method to develop further, especially since as at the present day only one open-sourced coupled frequency-domain model is available for analysing FOWT designs. These facts together make that the frequency-domain method is an interesting research topic, that could be of great value for the industry. The next section looks at some of the different FD models that have been applied for FOWT analysis, and will highlight why RAFT is deemed as the preferred tool to use for further development.

3.1.3. State of the art frequency-domain models

Multiple in-house frequency-domain models have been created, but all have their limitations. Hegseth and Bachynski [49] for example created a model specifically for SPAR-type designs, while as pointed out in section 2.6 currently about 80% of the commercial floating wind projects installed and in the pipeline plan to use a semi-submersible floater [83]. In the model by Lupton [70] the hydrodynamics are purely calculated via Morison's equation, which limits the the application to relatively small and slender structures [115]. Karimi, Buckham, and Crawford [65] developed a coupled FD tool that simplifies the control action. The model created by Pegalajar-Jurado, Borg, and Bredmose [93] is an example of a model that needs information that is pre-computed using time-domain tools.

To avoid the coupling between the FD hydrodynamics and prepossessed TD aerodynamics, research effort are increasingly focusing on control co-design [46]. This means that the complete system is optimized with all the components integrated in one software package, rather than designing each component sequentially on its own. However, to be able to do so all the parts of the system have to be linearized efficiently to be included in a FD method, including the aerodynamics and control. With the trend of increasing turbine sizes, which was found to be the case in section 2.3.4, the aerodynamic and hydrodynamic coupling as discussed in section 2.5.3 becomes even more important, making the work even more relevant. Work has been done to accurately linearize the aerodynamics and control so it can be included in such models, e.g. by de Souza, Hegseth, and Bachynski [25] and Lupton and Langley [71]. Improving on their work, the National Renewable Energy Laboratory (NREL) created the open-sourced software RAFT, that is capable of modelling the coupled hydrodynamic and aerodynamic parts of the system in the frequency-domain, including the control action, without the necessity of time-domain preprocessing. Therefore, RAFT is deemed the most state-of-the-art tool, with the perk that it is openly accessible. The workflow RAFT utilizes is described in the upcoming section.

3.1.4. RAFT's workflow

The following description of RAFT is retrieved from [46, 88]. RAFT represents a floating wind turbine system as a collection of different object types, which can be seen in figure 3.2. Each of these objects corresponds to a class in Python, and these classes interact following a set hierarchy. Some of these classes also interact with other external python models.

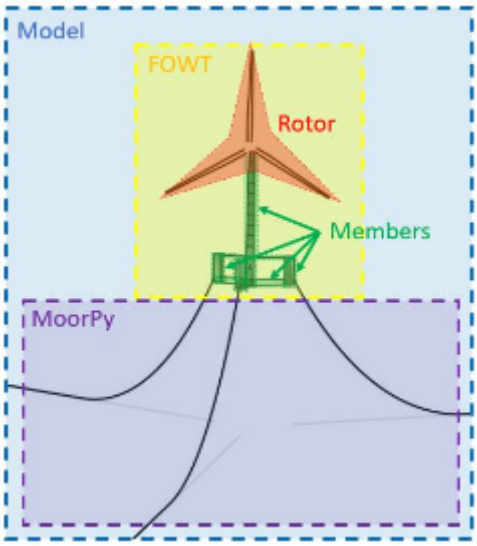

Figure 3.2: Object type collection as used in RAFT's model (retrieved from [88])

The model class contains the full RAFT representation of potentially an array of floating wind systems. Within it, the FOWT class describes the floater and turbine, while the mooring system is handled by a separate model: MoorPy [86]. Within the FOWT class, the floating platform and turbine tower are represented as a single rigid body composed of multiple member objects. A member is any circular or rectangular component of a FOWT which are assumed to be completely rigid objects with no flexibility, as RAFT currently does not support structural flexibility. By calculating the static properties of each member in the FOWT, it fills in its six-by-six mass and hydrostatic matrices. The class generates a mesh for each member that is used to calculate the hydrodynamic properties of each member through the use of strip theory by default. Optionally, external models or the Morison equation can be used for these properties. Further more, this class calculates the hydrodynamic damping and forcing, which are depending on the platform's position. It determines this position based on the mean aerodynamic forces determined using the rotor class and the mean hydrodynamic forces on one hand, and balancing this with the mooring and hydrostatic restoring forces on the other hand. The separate model used for to model the mooring system, MoorPy, models the mooring system in a quasi-static manner. Lastly, also the frequency dependent aerodynamic added mass, damping, and forcing matrices of the rotor are taken into consideration, used for determining motion responses in the model class.

In the rotor class, the rotor-nacelle assembly is described. It sets up attributes like the hub height, blade geometries, shaft tilt angle, hub wind speeds, rotor RPM, and blade pitch angles to create CCAirfoil and CCBlade objects [90]. The properties are used to calculate the distributed aerodynamic and control properties of the rotor. Using CCBlade, the class returns the loads and derivatives of the aerodynamics to extract the thrust and torque with respect to wind speed, rotor RPM, and blade pitch. The latter two are control dependent, meaning that RAFT can compute the overall aerodynamic effects of the system's rigid-body response, including the impact of the control system. The derivatives are used to determine the frequency dependent aerodynamic forces, added mass, and damping.

To better understand the working principles, the fundamentals of the frequency-domain method itself will be explained in the next section. It shows how the method typically deals with linearizing the equations of motion (EOM) for marine applications, both for floating objects as for the mooring system. Besides the methodology for marine applications, the section also shows the full derivation of the aerodynamic linearization.

3.2. Frequency-domain method

In this section the governing equations needed to determine the FOWT's motions using a frequency-domain method are presented. It will become clear how each main component, as mentioned in chapter 2, influence the total coupled equations of motion of the system. The general equation of motion for a system with six degrees of freedom in the time-domain is as given in equation 3.6:

$$\vec{F}(t) = (\mathbf{M} + \mathbf{A})\,\ddot{\vec{x}}(t) + \mathbf{B}\dot{\vec{x}}(t) + \mathbf{C}\vec{x}(t) \tag{3.6}$$

Herein the letters with an arrow above them represent column vectors, with $\vec{F}(t)$ being the resulting force vector, containing the forces inducing surge, sway, heave, roll, pitch and yaw motions. The actual motions themselves are represented in $\vec{x}(t)$, with the dots above representing time derivatives. Letters indicated in bold are 6-by-6 matrices, with **M** representing the mass and inertia, **A** the added mass, **B** the damping and **C** the stiffness matrices.

Note that equation 3.6 is in time-domain rather than in frequency-domain. Both methods solve the EOM of the system, but the approach in doing so differentiates compared to each other. Where time-domain methods directly integrate the EOM to obtain the system response as a function of time, the frequency-domain approach linearizes the system. Assuming a linearized system, the periodic motions can be expressed as being frequency dependent using the periodic function $ae^{i\omega t}$, with a the motion amplitude, and thus the first time derivative (velocity) is $i\omega a e^{i\omega t}$, and the second time derivative (acceleration) is $-\omega^2 a e^{i\omega t}$. This results in equation 3.7 for the general EOM in frequency-domain:

$$\vec{F}(\omega) = \left[-\omega^2\left(\mathbf{M} + \mathbf{A}(\omega)\right) + i\omega\mathbf{B}(\omega) + \mathbf{C}(\omega)\right]\vec{x}(\omega) \tag{3.7}$$

This linearized system makes it possible to determine the so-called response amplitude operators (RAOs), a frequency dependent factor representing the amplification of the floating platform's response (output) due to the force signal (input) [120]. For example, if the harmonic excitation force is due to an incoming wave described by $\zeta = \zeta_a e^{i\omega t}$, where ζ_a is the complex wave amplitude, the RAO would be:

$$\text{RAO}(\omega) = \frac{|a|}{\zeta_a} = \frac{F_0}{\mathbf{C}(\omega) - \left(\mathbf{M} + \mathbf{A}(\omega)\right)\omega^2 + i\mathbf{B}(\omega)\omega} \tag{3.8}$$

where $|a|$ represents the magnitude of the amplitude, and F_0 is the linear excitation force complex amplitude per wave height. Looking at this equation, it becomes clear that the responses are depending on the matrices **M**, **A**, **B**, and **C**, and the excitation vector. These themselves are consisting out of different parts, based on the components of a FOWT as mentioned in chapter 2. These parts can be calculating using different modelling techniques. All the the contributions can be superimposed if they are calculated to be acting around the same reference position [60]. The total matrices used for solving the EOM as used in this thesis consist out of:

$$\begin{aligned}
\mathbf{M} &= \mathbf{M} \\
\mathbf{A}(\omega) &= \mathbf{A}_{\textbf{potential}} + \mathbf{A}_{\textbf{aero}} \\
\mathbf{B}(\omega) &= \mathbf{B}_{\textbf{potential}} + \mathbf{B}_{\textbf{viscous}} + \mathbf{B}_{\textbf{aero}} \\
\mathbf{C}(\omega) &= \mathbf{C}_{\textbf{hs,body}} + \mathbf{C}_{\textbf{hs,mooring}} \\
\vec{F}(\omega) &= \vec{F}_{potential} + \vec{F}_{drift} + \vec{F}_{aero}
\end{aligned} \tag{3.9}$$

The meaning of these terms and how each of these terms are determined will be elaborated on in the next sections.

3.2.1. Mass and hydrostatics

The motions of the total system are partly determined by its hydrostatic properties. The hydrostatic response is due to forces that arise in the absence of fluid velocities and accelerations, i.e. the body has a relative velocity of zero with respect to the surrounding fluid. This subsection describes all the static properties that influence the total equations of motion as described in equation 3.7.

Mass and inertia

The mass matrix **M** is a direct result of the structural properties of the total system, i.e. the floater and the tower plus RNA:

$$\mathbf{M} = \begin{bmatrix} m & 0 & 0 & 0 & mz_G & -my_G \\ 0 & m & 0 & 0 & 0 & mx_G \\ 0 & 0 & m & my_G & -mx_G & 0 \\ 0 & -mz_G & my_G & I_{xx} & -I_{xy} & -I_{xz} \\ -mz_G & 0 & -mx_G & -I_{yx} & I_{yy} & -I_{yz} \\ -my_G & mx_G & 0 & -I_{zx} & -I_{zy} & I_{zz} \end{bmatrix} \tag{3.10}$$

Herein m is the total mass of the FOWT, x_G, y_G and z_G are the location of the centre of gravity in x, y and z direction respectively, and I_{ij} are the moment of inertia in DOF i due to a motion in DOF j. This matrix therefore purely depend on the dimensions and material properties of the total system.

Buoyancy restoring forces

The floater provides hydrostatic stiffness via its buoyancy in certain motion directions. In e.g. surge direction, the floater itself doesn't provide any stiffness as a force in that direction would just push the floater away. For other motions though, the floater's buoyancy will force the floater back to its equilibrium position as a result of the integration of the pressure field – in the absence of fluid velocities and accelerations – over the hull wet area. This describes the balance between hydrostatic and gravity loads, such that it is common to also include terms related to the body weight in the restoring matrix [23]. When assuming small motions, the hydrostatic stiffness matrix $\mathbf{C_{hs,body}}$ for a body with symmetry about the xz-plane is:

$$\mathbf{C_{hs,body}} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & c_{33} & 0 & c_{35} & 0 \\ 0 & 0 & 0 & c_{44} & 0 & 0 \\ 0 & 0 & c_{53} & 0 & c_{55} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{3.11}$$

With the restoring coefficients given by [32]:

$$c_{33} = \rho_w g A_{wp} \tag{3.12}$$

$$c_{35} = c_{53} = -\iint_{A_{wp}} x dx dy \tag{3.13}$$

$$c_{44} = \rho_w g V z_B - m g z_G + \iint_{A_{wp}} y^2 dx dy \tag{3.14}$$

$$c_{55} = \rho_w g V z_B - m g z_G + \iint_{A_{wp}} x^2 dx dy \tag{3.15}$$

where ρ_w is the water density, g is the gravitational constant, A_{wp} is the waterplane area, V the Floater's submerged volume in equilibrium position, and z_B and z_G are the vertical centers of buoyancy and gravity respectively. Note that the latter two terms include the weight trends, accounting for variations in the alignment between the upward buoyancy force and the downward gravity force due to pitching and rolling motions.

3.2.2. Hydrodynamics

Besides the hydrostatic response, the system's response is also influenced by its hydrodynamic characteristics, i.e. the subject now has a non-zero relative velocity with respect to the surrounding fluid. Depending on the size of the member of the floater (e.g. column, connecting pontoon, etc.) that is considered, different modelling approaches are considered as the most accurate. This subsection will explain these methods and when to apply which one, as well as how their results are influencing the total equations of motion as shown in equation 3.7.

Potential theory

Potential theory can used to find the added frequency dependent mass, damping, and wave induced excitation for a floating body, denoted by $x_{potential}$ in equation 3.7. For a large body with non-zero relative velocity with respect to the surrounding fluid, hydrodynamic loads not related with viscous effects can in general be accurately predicted by potential theory [23]. This solves the Navier-Stokes equations for a fluid in the presence of a free surface in a cartesian grid domain by using several assumptions. The fluid itself is considered as incompressible, irrotational, and inviscid. Furthermore it is assumed that local linearization can be applied. This allows for the introduction of a potential function Φ, whose spatial derivatives yield in the velocity in the direction of the derivative ($\nabla\Phi = \mathbf{u}$, where $\mathbf{u}$ is the velocity vector represented by $\mathbf{u} = (u, v, w)^T$). The pressure field in the fluid domain – and thus loads on the submerged structure – can then be obtained by applying Bernoulli's equation [32]:

$$\frac{\delta\Phi}{\delta t} + \frac{p}{\rho_w} + \left[\left(\frac{\delta\Phi}{\delta x}\right)^2 + \left(\frac{\delta\Phi}{\delta y}\right)^2 + \left(\frac{\delta\Phi}{\delta z}\right)^2\right] + gz = 0 \tag{3.16}$$

As the velocity potential is in the of the form $\Phi = \Phi_0(z)\sin(kx - \omega t)$, the resulting force will also be periodic with frequency ω [120].

The potential function itself is found by solving Laplace's equation subjected to boundary conditions (BCs) [99]:

- The no-penetration BC at the floating body surface, which implies the impermeability of the body's surface.
- The kinematic BC at the free surface and at the bottom, which implies that a particle at the free surface remains at the free surface, and that implies the impermeability of the bottom's surface, respectively.
- The dynamic BC at the free surface, which ensures that the pressure at the free surface is equal to the atmospheric pressure.
- The far field BC sufficiently far away of the floating body, so that the solution is know at locations where the influence of the body becomes negligible.

In practice, the problem is solved by applying a perturbation approach, expanding Φ as a series of functions with increasing order [23, 99]:

$$\Phi = \Phi_1\epsilon + \Phi_2\epsilon^2 + ... \tag{3.17}$$

where ϵ represents the nondimensionalized wave amplitude. It is used to describe the steepness of the wave, where a steeper wave behaves more nonlinear [99]. In general, loads associated with Φ_1 and Φ_2 are the most relevant for problems involving large floating bodies [23]. The determination of Φ_1 and Φ_2 will be explained next.

First-order wave loads

The parameter Φ_1 in equation 3.17 is associated with the first-order, linear wave loads. When viscous effects are not important and under the assumption of low wave amplitudes and small motions, loads obtained from Φ_1 are in general accurate for typical floating structures. The problem can be divided into the radiation and the diffraction problem [23].

The radiation problem assumes that the body is forced to oscillate in calm waters, at a given frequency ω, and thereby radiating waves to the surrounding fluid away from itself [60]. As a reaction to these motions, the fluid exerts forces and moments at the body, with components proportional to the body's acceleration and velocity [23]:

$$F_{radiation,i}(\omega) = -a_{ij}(\omega)\ddot{x}_j - b_{ij}(\omega)\dot{x}_j \tag{3.18}$$

where a_{ij} and b_{ij} are the added mass and radiation damping coefficients, respectively, and are gathered in the six-by-six matrices $\mathbf{A}_{\mathbf{potential}}$ and $\mathbf{B}_{\mathbf{potential}}$. The indices ij indicate a load in DOF i due to a unitary acceleration or velocity in DOF j.

In the diffraction problem the body is assumed as fixed in space, and subjected to incident waves. The potential is assumed as the superposition of the undisturbed wave and diffracted wave. The diffraction loads are the result of the undisturbed pressure field (Froude-Krylov) and wave scattering[60]. The

resulting pressure field over the body surface provides the wave excitation force in each DOF, which are gathered in the vector $\vec{F}_{potential}(\omega)$.

Second-order wave loads

The parameter Φ_2 in equation 3.17 is associated with the second-order wave loads. In the case of steeper waves, it is essential to also consider these higher-order terms to accurately estimate both loads and motions. By taking these terms into consideration, the boundary conditions are solved with a smaller error. Therefore, by applying the second-order potential theory, it is assumed that all the terms in the velocity potential, fluid pressure and wave loads are linear with the wave amplitude or proportional to the square of the wave amplitude [99]. The method produces mean forces as well as forces oscillating with both difference (with frequency $\omega_i - \omega_j$)- and sum (with frequency $\omega_i + \omega_j$) frequency oscillations, in addition to the linear solution. The mean forces contribute to the mean drift force $\vec{F}_{drift}$, while the difference- and sum frequency components lead to excitation outside the ranges of the wave excitation frequency. Low-frequency motions in surge, sway and yaw and are therefore important in the mooring system analysis [23]. For large FOWTs, roll and pitch can also be excited since their natural periods can become relatively long. Sum frequency forces can excite heave motions of TLPs [6].

The difference- and sum frequency forces can be determined using quadratic transfer functions (QTFs), which relate both difference- and sum frequency forces to pair of waves with unitary amplitude. However, the calculation of QTFs is time-consuming because it needs detailed discretization of the hull and he surrounding free-surface. A practical approximation is to neglect the free-surface forcing terms in the evaluation of Φ_2, and thereby omitting the need of meshing the free-surface, with significant reduction in computational time and modelling efforts [23]. This approximation has proven to be very accurate in representing the full QTF for semi-submersible FOWTs [18, 107].

Another approximation that can be applied for the difference frequencies is Newman's approximation [84]. In that case, the solution of the second order potential is not needed, as the method consists of using elements from the QTF's main diagonal to approximate elements closer to the diagonal. Therefore one can imagine that the method is more accurate for QTFs whos elements do not vary too much around the main diagonal. The method could prove useful for platforms with long natural periods of around 100 seconds or longer, since in that case difference frequencies resulting in loads with shorter periods are farther from the main diagonal [23].

Software

Software used to determine the just described linearized loads are called Boundary Element Methods (BEM). BEM normally finds its solutions by assuming a distribution of source functions over the body surface, which is discretized in panels. Together with with the boundary conditions, the potential functions associated with each panel form a system of equations that can be solved. An example of a floating body discritized in panels is given in figure 3.3.

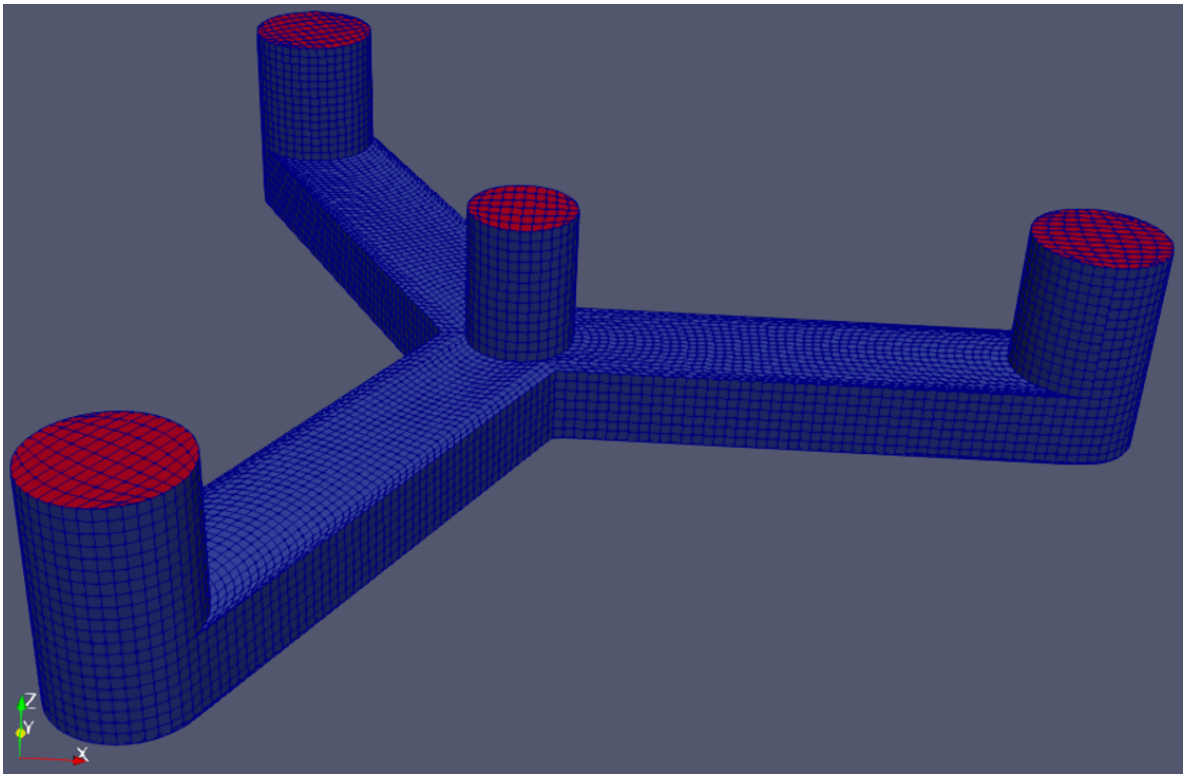

Figure 3.3: An example of the floating body of VolturnUS-S discritized in panels to apply BEM software, visualized in Rhino

Viscous forces and damping

The above mentioned potential theory is valid for large bodies, i.e. very large dimensions compared to the wavelength. However, for structures as semi-submersibles the perfect fluid model is not sufficient to fully describe the structure's hydrodynamic behaviour, since the structures also contain small elements, as e.g. can be seen in figure 2.21 for the OC4semi design. Typically, the elements are considered as small when their diameter or other projected cross-sectional dimension of the member in the wave direction, D, is significantly smaller than the wavelength, λ, yielding in the condition $\frac{D}{\lambda} < 0.2$ [26]. The flow around these elements is highly separated and induce nonlinear viscous effects that must be taken into consideration [23]. Other methods can be used to study this type of flow (e.g. model tests or CFD calculations), but they require a lot of computation time [51]. Results of such methods would yield in the additional damping matrix $\mathbf{B}_{\text{viscous}}$.

Morison's equation

In order to still use the linear theory, the viscous forces and damping can also be estimated using Morison's equation, a semi-empirical equation for the hydrodynamic forces on an object along the direction of incoming waves, neglecting the diffraction and radiation damping due to the slenderness of the considered object. For a cylinder in a current or with forward velocity, the Morison equation per unit length is [51]:

$$F_{Morison} = \underbrace{(1+C_a)\rho_w S\dot{u}}_{\text{added mass}} - \underbrace{C_a\rho_w S(\dot{u}-\ddot{x})}_{\text{Froude-Krylov}} + \underbrace{\frac{1}{2}\rho_w C_d D|u-\dot{x}|(u-\dot{x})}_{\text{viscous drag}} \tag{3.19}$$

where C_a and C_d are the added mass and drag coefficients respectively, ρ_w the water density, S and D are the section and cylindrical diameter. The fluid's particle velocities are denoted by u, and x represent the structure's rigid body motion, and for both holds that each dot above them represents one time derivative function.

Part of the right hand side (RHS) of equation 3.19 is linear and consists of inertial loads. The inertial loads are twofold, one term that is proportional with the water particle's acceleration, the added mass term, and one term which is proportional to the acceleration of the corresponding DOF, the Froude-

Krylov forces. Since the inertial loads are linear, they can directly be included in the complete FD model after being rotated and translated from the member's frame of reference to the floater's reference frame [115].

The viscous drag term however, consists out of a non-linear part. It can be rearranged as an excitation force and a damping force, both functions of the relative velocity of the fluid with respect to the corresponding DOF, as shown in equation 3.20 and 3.21. In this case it is assumed that the current velocity does not significantly contribute on the varying hydrodynamic forces. Therefore, assuming that wave particle velocities dominates, the linearized coefficient is introduced in equation 3.21 [29]. It makes the viscous drag force linear, and an iterative procedure is needed to solve the complete FD system [51].

$$F_{viscous,drag} = \gamma(u - \dot{x})u - \gamma(u - \dot{x})\dot{x} \tag{3.20}$$

$$\gamma(u - \dot{x}) = \begin{cases} f_v \frac{8}{3\pi} ||u - \dot{x}|| \rightarrow \text{Regular waves} \\ f_v \sqrt{\frac{8}{\pi}} \sigma_{u-\dot{x}} \rightarrow \text{Irregular waves} \end{cases} \tag{3.21}$$

with $f_v = 0.5\rho_w C_d D$, $||u - \dot{x}||$ denotes the relative velocity euclidean norm, and $\sigma_{u-\dot{x}}$ represents the standard deviation of the relative fluid velocity with respect to the corresponding DOF. The linearization of the Morison viscous drag term thus ends up in a set of two linearized forces, proportional to the fluid and to the corresponding DOF velocities respectively, as shown in equation 3.20. The damping matrix and the velocity force depend on all DOF motions, implying the FD solution to be solved through a fixed-point iterative process. This iterative method consists in setting an initial value of γ, e.g. 0, that will provide an initial solution and an updated γ. The same computation is carried out until a low error with respect to the previous solution is shown [51].

Note that it can be determined for each member of the floater whether the hydrodynamic properties are estimated using the potential flow solver or via Morison's equation, depending on their geometrical properties. Double calculations of members of the floater, i.e. applying both models to the same member, are to be avoided.

Mooring system

As mentioned in section 2.2, the mooring system is needed to avoid that the FOWT will drift away. In terms of the equations of motion this basically means that the mooring system ensures that part of the zeros in the hydrostatic stiffness matrix as shown in equation 3.11 become non-zero, emphasising the importance of the mooring system. In section 2.5, it was explained that the mooring system has non-linear behaviour. To linearize the mooring system so that the hydrostatic mooring stiffness $\mathbf{C_{hs,mooring}}$ can be determined for usage in the total EOM as given in equation 3.7, different approaches can be followed. Two optional methods will be described in this section.

Quasi-static approach

The first method uses classic catenary equations that are based on the water depth, floater position and the line's weight in water [32], and are called quasi-static modelling approaches. A catenary mooring line will exert a horizontal and vertical force on the floating structure, which are functions of the floater's offset. Just like a simple spring system, larger offsets will result in larger horizontal forces. However, if the floater moves, the mooring line will have axial elongation as well as overall geometric deformation. The mooring stiffness, $C_{hs,mooring}$, according to the quasi-static approach is a function of both axial stiffness, $C_{elastic}$, as geometric stiffness, $C_{geometric}$ [72]:

$$\frac{1}{C_{hs,mooring}} = \frac{1}{C_{elastic}} + \frac{1}{C_{geometric}} \tag{3.22}$$

Knowing the dimensions and material properties of the mooring system, static mooring line stiffness curves can be calculated. In contrast to the non-linear time-domain models, a linearized FD model provides only the fluctuating part of the solution [117]. Therefore the stiffness has to be linearized around the mean offset position. To determine this mean offset position, the offset where there is an equality between the static mooring line stiffness and the mean applied loads is searched for. At this mean offset, the mooring line stiffness can be linearized.

The quasi-static model however, doesn't account for the hydrodynamic loads on the mooring system. In order to solve these loads as well, the mooring system and line attachments must be linearized in a different manner. One such method is the lumped-mass method.

Lumped-mass approach
A lumped mass approach discretizes the dynamics of the mooring line over its length, by breaking the mooring line up into N evenly sized segments connecting $N+1$ node points. The line's mass is lumped at these node points, hence the name of the method. The nodes also account for the gravitational and buoyancy forces, hydrodynamic loads, and reactions from contact with the seabed. The hydrodynamic drag and added mass components of the segments are computed using Morison's equation [45]. The hydrodynamic Morison forces on mooring lines account for the relative motions between water particles and line sections [51]. The axial stiffness can be modeled by applying a linear stiffness to each line segment, in tension only, neglecting bending and torsional stiffnesses [45]. The reactions from contact with the seabed are represented by vertical stiffness and damping forces. Figure 3.4 shows a representation of a lumped-mass model.

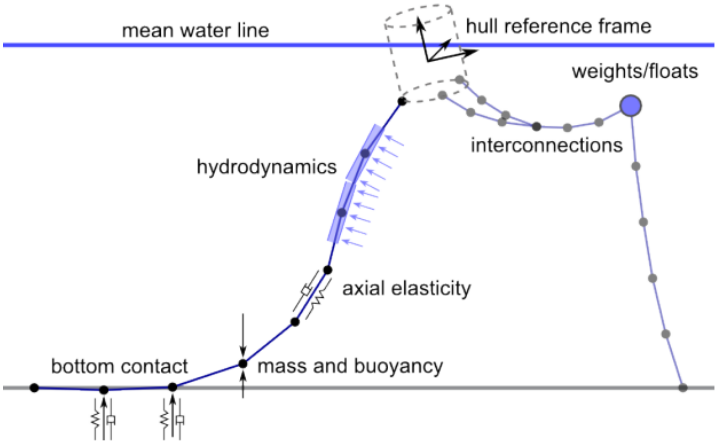

Figure 3.4: Lumped-mass mooring model elements (retrieved from [108])

A lumped-mass approach gives more accurate results of the mooring line tensions compared to a quasi-static approach for dynamic systems due to the fact that the hydrodynamics are taken into consideration, but thereby it also comes at higher computational costs. Quasi-static approaches can be sufficient for initial stage optimization, but e.g. a lumped-mass approach is needed for more accurate predictions of the mooring line tensions [72].

3.2.3. Aerodynamics and control

Now that it is clear how the hydrodynamic contributions in the EOM can be determined, the only contributions left to determine for the EOM as presented in equation 3.7 are the aerodynamic contributions. As mentioned in section 2.5, there is a strong coupling between the aerodynamics and the FOWT motions. How the aerodynamics, including the effects of FOWT motions and turbine control, can be linearized and implemented in the overall EOM will be explained in this section. The methodology is based on the work by Hall et al. [46], who expanded on the work of de Souza, Hegseth, and Bachynski [25] by adding generator torque control, turbulent wind excitation and nacelle-velocity feedback control. The method requires a Blade Element Momentum method (BEM*) to determine aerodynamic loads. Therefore, before explaining the linearization method, first the theory behind a BEM* method is explained.

Blade Element Momentum method

BEM* methods can be used to calculate the values of the aerodynamic excitation and load derivatives of the rotor. Therefore it uses some other theories, as will be briefly described next.

Actuator disc theory

The method is partly based on the actuator disc theory, wherein it is assumed that the affected mass of air that flows through the rotor disk area remains separate from the air flow which does not, and that no air flows across the boundary of this layer. This creates a so-called stream-tube as depicted in figure 3.5, in which the flow mass rate along the stream-tube will remain the same due to the assumptions made. Behind the turbine the stream-tube is expanded due to the energy extraction of the turbine, since it causes a wind velocity and pressure drop [73]. Using the assumption that the mass flow rate remains the same over the stream tube, the wind velocity at the turbine can determined as $U_w(1-a)$, with U_w the mean incoming wind speed, and a the axial component of the induced flow.

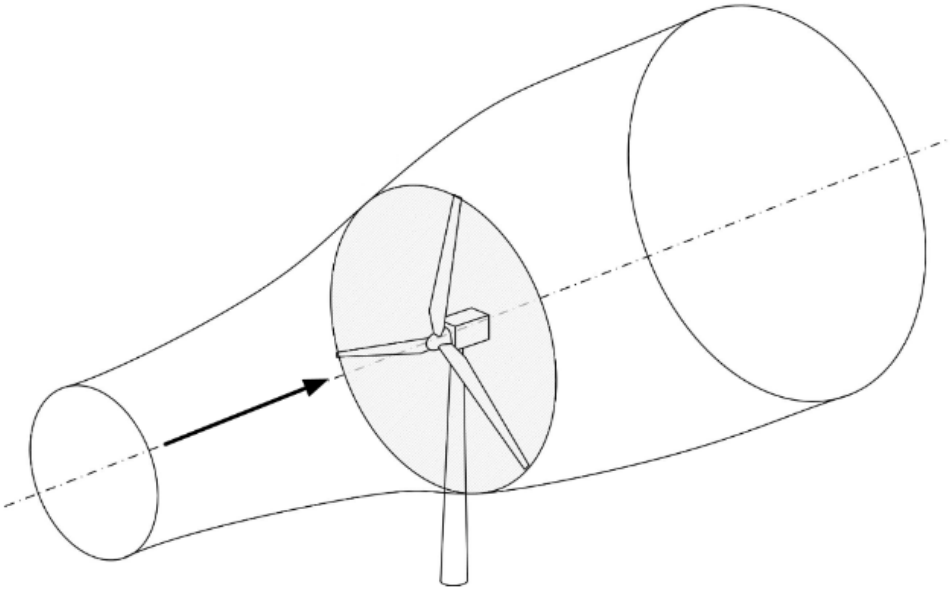

Figure 3.5: A stream-tube for a wind turbine, with the turbine located at the actuator disc (retrieved from [91])

Angular momentum theory

Furthermore the method applies the angular momentum theory. This states that applying a torque on a rotor disc by the air flowing trough the disc, necessitates and equal and opposite torque to be imposed upon the air. Consequently, the air behind the rotor, i.e. in the wake field, will rotate in the opposite direction of that of the rotor, and thus it gains angular momentum. The trajectory of an air particle in the wake field of a rotor disc is depicted in figure 3.6. The change in tangential velocity at the disc due to this can be expressed using the tangential component of the induced flow, a'. Assuming there is no such component in the mean incoming wind speed, and it is $2\Omega r a'$ downstream of the rotor in the opposite direction compared to the rotor's motion, it is given by [73]:

$$a' = \frac{a(1-a)}{\gamma_r^2} \tag{3.23}$$

where γ_r is the local tip speed ratio.

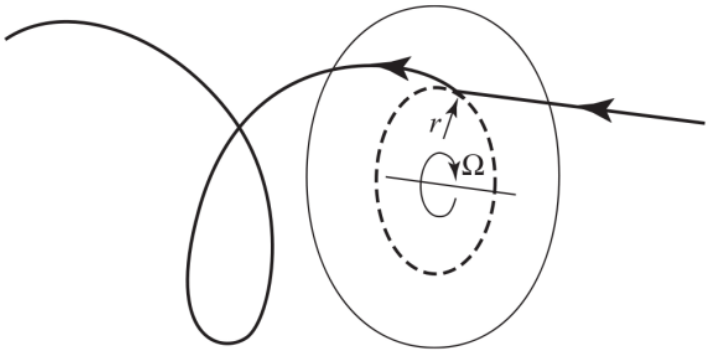

Figure 3.6: The trajectory of an air particle in the wake field of a rotor disc (retrieved from [73])

BEM* theory

The BEM* method is based on these principles of conservation of mass and momentum, based on the assumptions of steady, inviscid, incompressible flow without radial and circumferential dependency [58]. The forces on the rotor can be deduced from aerodynamic lift and drag on a blade section, with width δr, determined from two-dimensional airfoil theory or previously recorded experimental data. The discretization of the rotor blades is presented in figure 3.7.

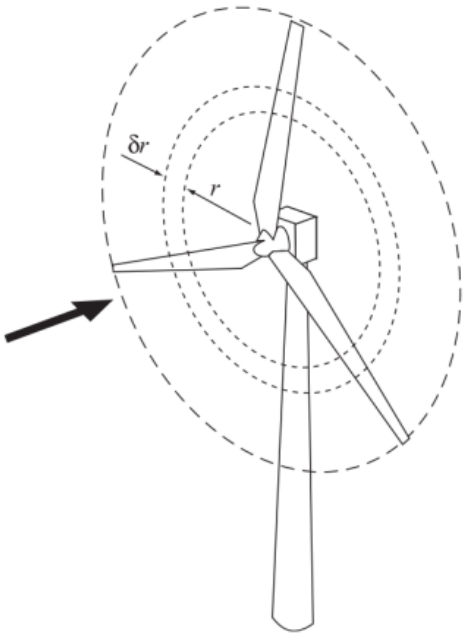

Figure 3.7: The blade segment δr visualized (retrieved from [73])

Using the two-dimensional airfoil theory as presented in section 2.3.2, the following equations for the thrust and torque are found, for N blade elements:

$$dT = \frac{1}{2}\rho_a c W^2 N(c_L cos(\alpha) + c_D sin(\alpha))dr \tag{3.24}$$

$$dQ = \frac{1}{2}\rho_a c W^2 N(c_L sin(\alpha) - c_D cos(\alpha))rdr \tag{3.25}$$

in which r is the radial distance of segment N, and the apparent wind speed W was:

$$W = \sqrt{U_w^2(1-a)^2 + \Omega^2 r^2(1+a')^2} \tag{3.26}$$

Integrating over the blades to obtain the total thrust and torque therefore still have two unknown quantities: The axial and circumferential velocity at the rotor disc. To compute these, two additional

equations for thrust and torque on the blade sections are used, based on the conservation of axial and tangential momentum within a stream tube [120]:

$$dT = 4\pi\rho_a W^2(1-a)ardr \tag{3.27}$$

$$dQ = 4\pi\rho_a W\Omega(1-a)a'r^3dr \tag{3.28}$$

With these two additional equations, a solution can be obtained in an iterative manner.

Corrections to BEM* theory

To improve the accuracy of the BEM* method, additional corrections can be applied. One of the major limitations of the BEM* method is that there is no influence of vortices shed from the blade tips into the wake of the induced velocity field [81]. Therefore the method is often expanded to take the effect tip losses in consideration by the use of correction factors [13]. A well known model for this is the Prandtl correction factor. This correction factor is used to modify the momentum part of the BEM* theory, equations 3.27 and 3.28. The effect on induced velocity in the rotor plane is most pronounced near the tips of the blades, an area that also has the greatest influence on the power produced by the turbine [81]. The Prandtl correction factor therefore is a function of the normalized radius of the blade, and becomes more distinct near the tip of the blade.

Similar to the tip-loss model, the hub-loss model is employed to adjust the induced velocity caused by the shedding of a vortex near the rotor's hub. The hub-loss model utilizes an implementation closely resembling the Prandtl tip-loss model to characterize the impact of this vortex. The segment can be affected by both tip and hub losses, in which case the total loss factor is determined by the multiplication of both [81].

The BEM* theory faces a limitation when the induction factor a exceeds approximately 0.4. This situation arises in high TSR scenarios, leading to a turbulent wake state. In this state, the flow in the far wake of the turbine starts propagating upstream, violating the assumptions made for BEM* theory. Although physically such a flow does not occur, Glauert introduced a correction factor for the rotor thrust coefficient, compensating for increased flow entrainment from outside the wake and heightened turbulence. The method was developed to determine a correction factor for the total rotor, but can also be applied for for local coefficients of blade segments. However, because of this, it is important to understand the relationship between the Glauert- and tip-loss correction models. E.g. when the losses near the tip are high, the velocities are large, and therefore the possibility of a turbulent wake near the tip increases. Thus, the total induced velocity calculation for each segment must use a combination of both corrections [81].

An additional limitation of the BEM* method can occur when the turbine operates at yaw angles with respect to the incoming wind, and as a result experience a skewed wake behind the rotor. The BEM* method considers axis-symmetric flow, and therefore do not take this skewed wake into account. To deal with this the skewed wake correction can be applied, although it has some limitations. There is no firm theoretical background for applying the correction factor in combination with the BEM* method. Applying the correction factor was found to improve results for predicting yaw moments and motions compared to the BEM* method without the correction factor in some cases, but the correction might be to large in other situations [81].

In most software programs it can be chosen whether the correction factors are included or not. Some software packages will be highlighted in the next subsection.

Software

Some well known software packages that apply the BEM* method are NREL's OpenFAST [87] or Qblade, developed by the technical university of Berlin [78]. Another option, used in this thesis, is the steady-state CCBlade. The latter is based on the work done by Ning [85], who mathematically proofed that the applied solution algorithm has a superlinear convergence rate, always converges, and is continuously differentiable. This makes that the method is fast, robust, and smooth.

BEM* software can be used to determine steady-state rotor power and torque, along with their derivatives with respect to the blade pitch angle, wind speed, and rotor speed, all azimuthally averaged. The steady-state quantities are used in calculating the mean system loads and e.g. the annual energy production. Their derivatives can be used in computing the aerodynamic contributions to the system's dynamics in the frequency-domain. How the latter is done for the FD method is explained next.

Aerodynamic added mass, damping and excitation vector

The following method is as described in the work by Hall et al. [46]. To write the influence of the rotor aerodynamics and control into the total equations of motion as shown in equation 3.7, the thrust is linearized using a Taylor series expansion, keeping only the first derivative term. Knowing that the generated thrust depends on the relative wind speed at hub height, W, the rotor speed, Ω, and the blade pitch angle, β, the linearized thrust T after applying the Taylor expansion becomes as given in equation 3.30. Notice that the rotor speed, Ω, and blade pitch angle, β, are two controllable variables, meaning that the controller is taken into consideration.

$$T = T_0 + \frac{\delta T}{\delta W}\Delta W + \frac{\delta T}{\delta \Omega}\Delta\Omega + \frac{\delta T}{\delta \beta}\Delta\beta \tag{3.29}$$

Herein T_0 is the initial thrust, i.e. the first term of the Taylor expansion, ad W is the apparent wind speed.

Knowing that $\Delta W = W - U_w = -\dot{x}$, where U_w is the mean incoming wind speed, means that the relative wind speed at hub height can be rewritten in terms of x and U_w, assuming a constant mean incoming wind speed, resulting in:

$$T = T_0 + \frac{\delta T}{\delta U_w}\Delta(U_w - \dot{x}) + \frac{\delta T}{\delta \Omega}\Delta\Omega + \frac{\delta T}{\delta \beta}\Delta\beta \tag{3.30}$$

The same exercise can be done for the aerodynamic torque Q:

$$Q = Q_0 + \frac{\delta Q}{\delta U_w}\Delta(U_w - \dot{x}) + \frac{\delta Q}{\delta \Omega}\Delta\Omega + \frac{\delta Q}{\delta \beta}\Delta\beta \tag{3.31}$$

In addition, the rotor dynamics are as given in equation 3.32, describing how the rotational motion of the rotor changes for external torques applied to the rotor.

$$I_r\dot{\Omega} = Q - N_g(\tau_{g,0} + \Delta\tau_g) \tag{3.32}$$

where I_r is the rotor's mass moment of inertia, N_g is the gear ratio and τ_g is the electrical generator torque, with the latter consisting of a mean torque $\tau_{g,0}$ and a torque variation $\Delta\tau_g$. In equilibrium $Q_0 = N_g\tau_{g,0}$, so that substituting equation 3.31 in 3.32 yields:

$$I_r\dot{\Omega} = \frac{\delta Q}{\delta U}\Delta(U_w - \dot{x}) + \frac{\delta Q}{\delta \Omega}\Delta\Omega + \frac{\delta Q}{\delta \beta}\Delta\beta - N_g\Delta\tau_g \tag{3.33}$$

The only parameters that need further clarification are the controlled blade pitch angle and generator torque. These – of course – depend on the control strategy adapted. As explained in section 2.4, generally the generator torque is PI controlled based on the rotor speed offset in the below rated control regime:

$$\Delta\tau_g = k_{P\tau}\Delta\Omega + k_{I\tau}\int \Delta\Omega dt \tag{3.34}$$

where $k_{P\tau}$ is the generator torque proportional gain and $k_{I\tau}$ the generator torque integral gain.

In the above rated regime the turbine is blade-pitch controlled. Similarly to the generator torque control linearization, the blade pitch control action can be linearized. Besides the proportional and integral gains, which effectively would be a VSVP controller if used on their own, other control strategies such as e.g. mentioned in section 2.4.3 can be adapted. If, for example, a nacelle-velocity-feedback is implemented, the linearized blade pitch becomes:

$$\Delta\beta = k_{P\beta}\Delta\Omega + k_{I\beta}\int \Delta\Omega dt + k_{Px}\dot{x} \tag{3.35}$$

Now, equations 3.30 and 3.33 can be rewritten using the linearized control equations 3.34 and 3.35, noting that $\Delta\Omega = \dot{\phi}$, where ϕ is the rotor azimuth angle:

$$T = T_0 + \frac{\delta T}{\delta U_w}U_w + \left(k_{Px}\frac{\delta T}{\delta \beta} - \frac{\delta T}{\delta U_w}\right)\dot{x} + \left(\frac{\delta T}{\delta \Omega} + k_{P\beta}\frac{\delta T}{\delta \beta}\right)\dot{\phi} + K_{I\beta}\frac{\delta T}{\delta \beta}\dot{\phi} \tag{3.36}$$

$$I_r\ddot{\phi} - \left(\frac{\delta Q}{\delta\Omega} + k_{P\beta}Q_\beta - N_g k_{P\tau}\right)\dot{\phi} - \left(k_{I\beta}Q_\beta - N_g k_{I\tau}\right)\phi = \frac{\delta Q}{\delta U_w}U_w + \left(k_{Px}\frac{\delta Q}{\delta\beta} - \frac{\delta Q}{\delta U_w}\right)\dot{x} \tag{3.37}$$

Assuming harmonic oscillation, these equations can be written in the frequency domain. For e.g. ϕ we can than write that $\phi = \phi(\omega)$, $\dot{\phi} = i\omega\phi$ and $\ddot{\phi} = -\omega^2\phi$. Applying this principle to equation 3.37 and rewrite it so that a transfer function between $\phi(\omega)$ and $U_w(\omega)$, $x(\omega)$ is established, we get:

$$\phi(\omega) = \frac{\frac{\delta Q}{\delta U_w}U_w(\omega) + \left(\frac{\delta Q}{\delta\beta}k_{Px} - \frac{\delta Q}{\delta U_w}\right)i\omega x(\omega)}{-I_r\omega^2 - (\frac{\delta Q}{\delta\Omega} + \frac{\delta Q}{\delta\beta}k_{P\beta} - N_g k_{P\tau})i\omega - (\frac{\delta Q}{\delta\beta}k_{I\beta} - N_g k_{I\tau})} \tag{3.38}$$

This relationship can be substituted in equation 3.36, so that the following equation for frequency-dependent variations in rotor thrust is found:

$$T(\omega) = \frac{\delta T}{\delta U_w}U_w(\omega) - \left(\frac{\delta T}{\delta U_w} - k_{Px}\frac{\delta T}{\delta\beta}\right)i\omega x(\omega) - H_{QT}(\omega)\left[\frac{\delta Q}{\delta U_w}U_w(\omega) - \left(\frac{\delta Q}{\delta U_w} - k_{Px}\frac{\delta Q}{\delta\beta}\right)i\omega x(\omega)\right] \tag{3.39}$$

where $H_{QT}(\omega)$ is the transfer function from rotor torque to thrust:

$$H_{QT}(\omega) = \frac{\left(\frac{\delta T}{\delta\Omega} + k_{P\beta}\frac{\delta T}{\delta\beta}\right)i\omega + k_{I\beta}\frac{\delta T}{\delta\beta}}{\omega^2 I_r + i\omega\left(\frac{\delta Q}{\delta\Omega} + k_{P\beta}\frac{\delta Q}{\delta\beta} - N_g k_{P\tau}\right) + k_{I\beta}\frac{\delta Q}{\delta\beta} - N_g k_{I\tau}} \tag{3.40}$$

From this, the fore-aft terms for aerodynamics added mass, damping, and turbulent wind excitation can be extracted for the turbine including the control action:

$$\begin{aligned}
a_{aero}(\omega) &= \mathfrak{Re}\left[\frac{1}{i\omega}\left[\frac{\delta T}{\delta U_w} - k_{Px}T_\beta - H_{QT}(\omega)\left(\frac{\delta Q}{\delta U_w} - k_{Px}\frac{\delta Q}{\delta\beta}\right)\right]\right] \\
b_{aero}(\omega) &= \mathfrak{Re}\left[\frac{\delta T}{\delta U_w} - k_{Px}T_\beta - H_{QT}(\omega)\left(\frac{\delta Q}{\delta U_w} - k_{Px}\frac{\delta Q}{\delta\beta}\right)\right] \\
\hat{f}_{aero}(\omega) &= \left(\frac{\delta T}{\delta U_w} - H_{QT}(\omega)\frac{\delta Q}{\delta U_w}\right)U_w(\omega) = H_{Uf}(\omega)U_w(\omega)
\end{aligned} \tag{3.41}$$

These hub-height quantities are then transformed to be about the platforms reference point, accounting for their coupled affect on surge and pitch in the final aerodynamic added mass matrix, $\mathbf{A}_{aero}(\omega)$, damping matrix, $\mathbf{B}_{aero}(\omega)$, and excitation vector, $\vec{F}_{aero}(\omega)$. These can than be included in the system's total EOM, as given in equation 3.7.

3.3. RAFT's results and potential improvements

The above described linearization method is implemented in RAFT. This section shows the aerodynamic results as found by Hall et al. [46] by using RAFT, as well as the total found system's response. Based on these responses, potential improvements are identified.

3.3.1. Aerodynamic coefficients

If the aerodynamic linearization method as described in section 3.2.3 is applied on IEA's 15 MW reference turbine with RAFT's software, the coefficients shown in figure 3.8 are found. However, due to an increpancy in the software, the nacelle velocity feedback gain is not included, only the blade-pitch and generator-torque control settings as designed for the UMaine VolturnUS-S reference design are included in this calculation (see [3] for the design report). Three wind speeds are below rated and three are above rated, showing the large effect of wind speed on the coefficients. Furthermore the figure shows the corresponding rotor-averaged wind speed amplitudes calculated for the turbine's rotor area with class IB-NTM turbulence levels [46].

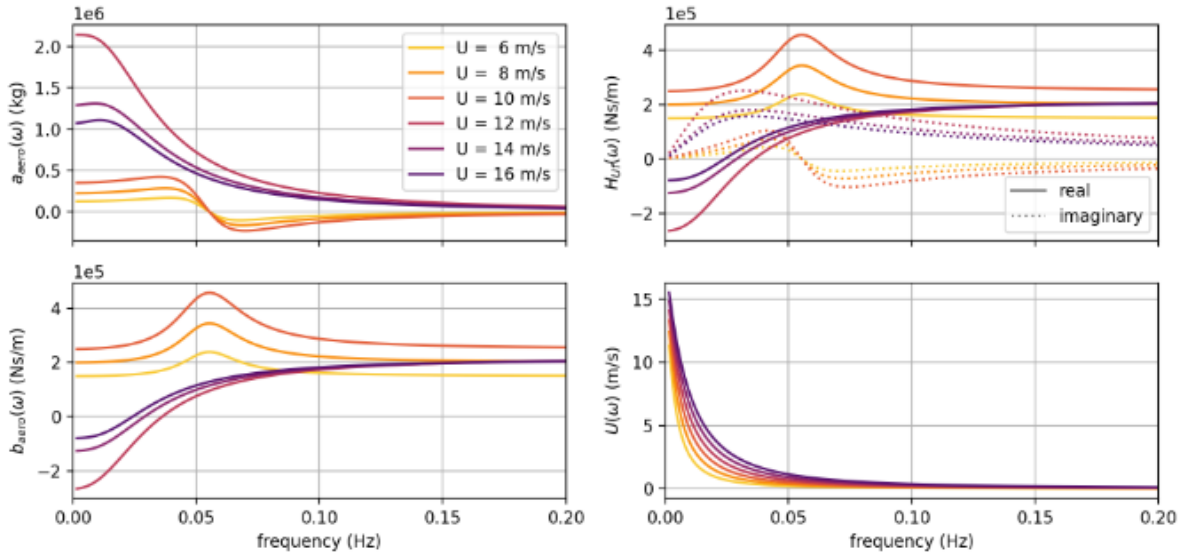

Figure 3.8: Calculated aerodynamic added mass, damping, and excitation coefficients and rotor averaged wind speed amplitudes for the IEA 15-MW Reference turbine over three below and three above rated wind speeds (Retrieved from [46])

The two figures on the right multiplied by each other equal the aerodynamic excitation, as can be seen in equation 3.41, where the top right figure represent the transfer function between the wind speed and aerodynamic excitation, and the bottom right figure represents a Kaimal wind spectrum. The results in the figure make clear that the dynamic response differs over different wind speeds, since the aerodynamic added masses, damping and excitation coefficients differ, impacting the responses as known by equation 3.7. The figure shows the negative damping for above rated wind speeds for a certain frequency range, which can lead to the known instabilities as mentioned in section 2.4.1. The below rated wind speeds have a frequency range where the added masses becomes negative. This leads to shorter decay periods, which aligns with experimental results found by e.g. Bachynski et al. [5] and Goupee, Kimball, and Dagher [38].

3.3.2. Total responses

The aerodynamic results are included in the total system response determination as well. Hall et al. [46] applied RAFT to the OC3 Hywind spar [59], the OC4-DeepCwind semi-submersible [100], and the VolturnUS-S semi-submersible [3] in order to demonstrate and validate RAFT. The following section describes the conclusions as found by the authors in that report. The conclusions are drawn based on the results from RAFT that are compared against either reference values or results from equivalent OpenFAST time-domain simulations. The three tested designs are given in figure 3.9, and for scale reference, the OC3 and OC4 FOWTs are designed for the open-sourced 5 MW turbine with a hub-height of 90 meters, while the VolturnUS-S design is designed for the open-sourced 15 MW turbine with a hub-height of 150 meters.

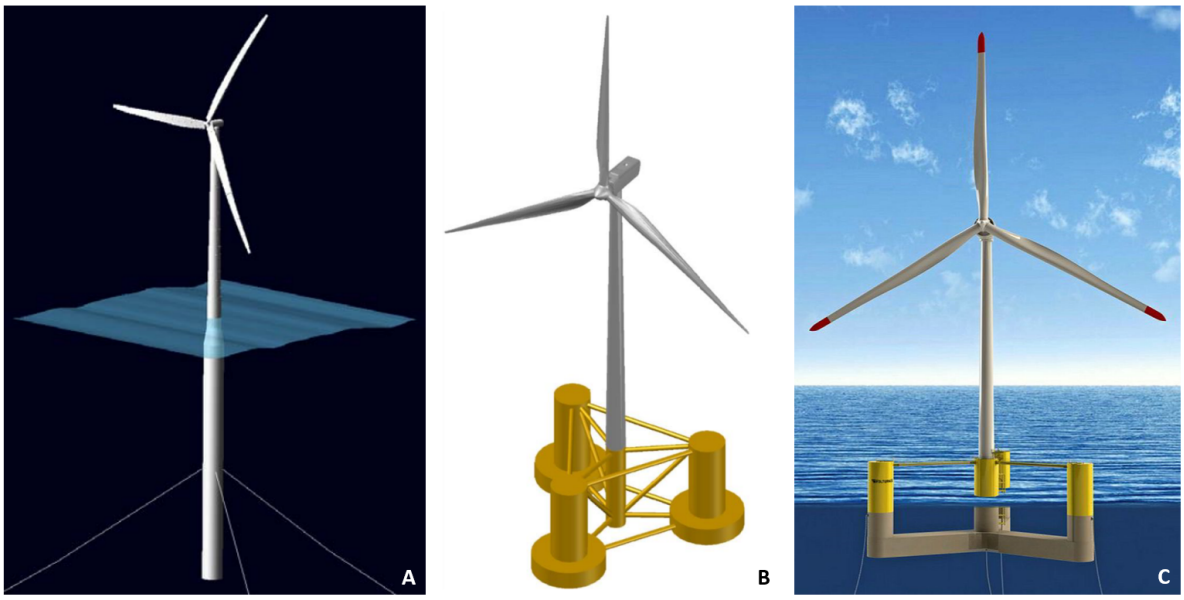

Figure 3.9: A visualization of the three designs tested in RAFT. A: The OC3 Hywind Spar [59], B: The OC4-DeepCwind semi-submersible [94], C: The VolturnUS-S semi-submersible [3]

In order to match the static properties of the FOWT designs in RAFT with the values as reported in the technical reports of the designs, such as the overall mass, mass distribution of the system, and system inertias, the models are put in RAFT using backwards engineering. During this, the natural frequencies were also matched making use of the computed stiffness and inertia matrices. Difficulties arose in matching the yaw inertias during this process, and as a result some problems arose for matching the yaw natural periods.

The total response was determined for a case with wind and irregular waves. The power spectral density responses for surge, heave, and pitch motions, under steady 8 meters per second winds and a JONSWAP wave spectrum with a 12 second peak period and 6 meter significant wave height, are shown in figure 3.10, both obtained by RAFT as well as by the higher fidelity time-domain program OpenFAST. Note that the time-domain response is less smooth, due to the time-trace results being transformed into spectral responses using a Fast Fourier Transform (FFT).

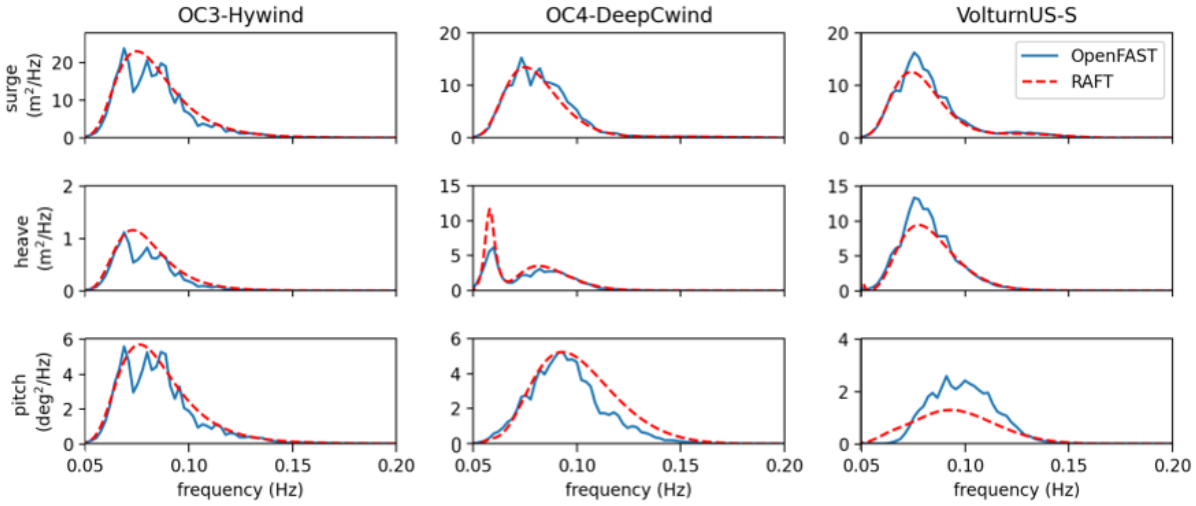

Figure 3.10: Power spectral densities of surge, heave and pitch response for three FOWT designs to irregular waves and steady wind as found by using RAFT and OpenFAST (retrieved from [46])

Although the overall results show good levels of agreement between the frequency-domain method and the higher fidelity time-domain method, there are some notable differences, which seem to appear

where diffraction and radiation effects would be greatest. The first one is that RAFT overpredicts the heave response of the OC4-DeepCwind semi-submersible at 0.06 Hertz. The author attributed this to the fact that RAFT relies on 2D strip-theory for the hydrodynamic calculations, whereas OpenFAST uses the 3D potential flow-theory as described in section 3.2.2. RAFT therefore does not model the transverse dynamic pressure loads, resulting in less damping at the frequencies where the impact of the heave plates would be the greatest. This is the case for motions in the vertical direction and at lower frequencies, which is where the largest difference were recognized. The second notable difference in the comparison between the results is the pitch response of the VolturnUS-S design. In figure 3.10 it is seen that the pitch response as predicted by RAFT for this design has a lower peak but extends into the lower frequency ranges compared to the OpenFAST results. This typically indicates that the damping is underestimated in the lower frequency range and excitation is missing at the peak. The author's reasoning for this noticed difference is similar to the reasoning behind the difference found for the OC4-DeepCwind design. Just as the heave plates of that design, the diffraction and radiation effects of the large pontoons between the columns for the VolturnUS-S semi-submersible design are not modelled accurately in RAFT. The pontoons could experience heave and pitch excitation from the transverse dynamic pressure loads, that are currently modelled in OpenFAST but not in RAFT. Furthermore, the pontoons would cause wave radiation damping, which is not captured in RAFT's approach.

Besides these motion response errors, the largest differences are found in the upwind mooring line fairlead tensions, which are shown in figure 3.11. As mentioned in section 3.1.4, RAFT applies a quasi-static mooring model. This model does not include hydrodynamic effects, as explained in section 3.2.2. The neglected drag and inertia effects of the mooring line become significant at higher platform motions, explaining the differences as seen in figure 3.11, as the OpenFAST results applied a lumped-mass approach. How the mooring line dynamics are included in such an approach is also explained in section 3.2.2. To demonstrate the differences between the modelling techniques, OpenFAST was rerun with a quasi-static mooring line model, resulting in the green dotted lines in figure 3.11. For that case, the tensions agree very well with RAFT's predictions.

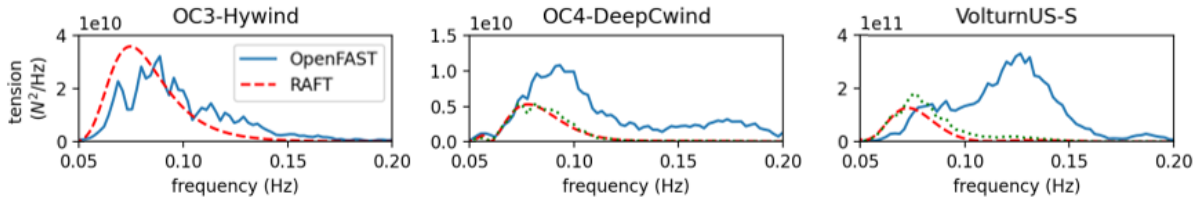

Figure 3.11: Power spectral densities of mooring tension for three FOWT designs to irregular waves and steady wind as found by using RAFT and OpenFAST, with the latter applying the lumped-mass approach (blue lines) and a quasi-static approach (green dotted lines) (retrieved from [46])

Furthermore Hall et al. [46] showed that there are mispredictions in the nacelle acceleration and tower-base fore-aft bending moments for all the floaters. However, the deviations found compared to the time-domain simulation results are consistent with the differences observed in platform motions, implying that better modelling of the platform motions results in better predictions for these quantities.

3.3.3. Potential improvements

In the previous section differences between results found by RAFT and the higher-fidelity OpenFAST are shown. The explanations given for these differences embody some of the potential improvements of the software. As the differences were identified to be for certain motions within frequency ranges where the diffraction and radiation effects would be the greatest, the software could be improved by applying a better hydrodynamic model. RAFT currently models the hydrodynamics using 2D strip theory wherein these effects, as well as the transverse dynamic pressure loads, are not captured. So instead of the current model, a 3D potential flow model can be adopted to improve the modelling of motion responses.

Furthermore, the results highlighted that the mooring modelling method should be improved in order to better determine the mooring line tensions, especially for higher frequencies. The current quasi-static method does not include the hydrodynamic effects of the mooring line. To improve on this, e.g.

a lumped-mass approach could be implemented so that the mooring line inertia and drag loads are included according to the method as briefly explained in section 3.2.2.

Besides the potential improvements based on the found results from the previous section, some other potential improvements were found by diving into the applied code. The first one is that, as mentioned in section 3.3.1, the software did not implement the nacelle velocity feedback method correctly. Although the methodology as shown in section 3.2.3 is implemented in the code, the gain itself does not have a frequency dependency, and as a result the simulations were performed with the gain set to zero. Nevertheless they have published the results, as their primary goal was to create an open-sourced model to quickly evaluate integrated FOWT designs. The published code is version 1.0, implying it is under development.

Another noticeable potential improvement found by looking at the code is that the frequency dependent excitation is set to zero as well. Only the mean aerodynamic forces are applied to the system. RAFT is not yet capable of dealing with the turbulent wind spectrum in the total response. It is capable of determining the wind spectrum and the thereby following thrust fluctuations itself for the case without platform motions, as shown in figure 3.3.1 in the bottom right figure, but the aerodynamic forces caused by the platform motions are not included, as can bee seen in figures 3.10 and 3.11. It is assumed that the wind and waves are uncorrelated, so that it sums up the resulting power spectral densities due to the wave and wind induced excitations.

The latter two potential improvements also explain why the responses given in figures 3.10 and 3.11 are given starting from 0.05 Hertz, as the wind excitation lies mainly in the range below this frequency, as can be seen in the bottom right figure of figure 3.3.1. Since the feedback control is not implemented correctly, the response in this frequency range is expected to deviate from the case where it is implemented, which is the case in the TD software. To omit the wind excitation problems, the simulations are for a steady wind case.

Lastly, there is a potential improvement identified based on RAFT's workflow. As mentioned in section 3.1.4, RAFT does not take structural flexibility into consideration. For turbines with higher rated powers, which is the ongoing trend as shown in section 2.3.4, this topic becomes more important due to the higher towers and larger forces acting on the towers. Therefore, the assumption of rigid bodies becomes less suitable the larger the turbines get. One note to this however, is that both the FD as the TD approach results as shown in figures 3.10 and 3.11 assume rigid bodies, so there are no differences between these results due to this assumption.

The goal of this thesis is to improve on RAFT to create an improved, coupled frequency domain model for floating offshore wind turbines including aerodynamics, hydrodynamics and control. Ideally, one would like to address all the potential improvements mentioned, but due to time limitations not all potential improvements shall be tackled in this thesis. The next chapter outlines which of the identified potential improvements are addressed in this work.

4

Research plan

In section 3.3.3 multiple potential improvements that can be made to RAFT's software are identified. As the goal of this thesis is to create an improved, coupled frequency domain model for floating offshore wind turbines including aerodynamics, hydrodynamics and control, the focus in this work will be on improving the hydrodynamic predictions, including mooring, as well as on improving on the aerodynamic predictions. To improve the aerodynamics of the model, the main potential improvement was identified as being able to include the nacelle velocity feedback control. For improving the hydrodynamic responses, a better hydrodynamic model must be implemented, as well as a different mooring modelling method. This work will keep the rigid body assumptions for now, so no improvements will be made regarding taking structural flexibility into account.

In this chapter the main research question is stated, which is formed based on the potential improvements that will be addressed in this thesis. Some sub-questions are formed to aid answering the main question, just as (sub-)objectives are formed that need to be achieved in order to answer the research questions. During this research, the enhanced frequency-domain method will be evaluated using UMain's VolturnUS semi-submersible design in combination with IEA's 15 MW reference turbine, since this is the largest turbine out of the three already tested with RAFT. In section 2.3.4 it is demonstrated that the ongoing trend for offshore wind turbines is that they increase in size over the years, so the largest turbine is the most representative one for current designs. Furthermore, the coupling between the aerodynamics and hydrodynamics becomes more pronounced for larger turbines. As mentioned in section 3.1.2, the maritime industry is already experienced in using FD methods to solve the hydrodynamics, but coupling this knowledge with frequency dependent aerodynamics is new. So, the design with the most pronounced aerodynamics is deemed to be the most interesting one to investigate this coupling.

4.1. Research question

Based on the potential improvements that the state-of-the-art open-sourced frequency-domain tool RAFT still has will be addressed in this thesis, the following research question is formed:

> *How effective is the improved coupled frequency domain model for floating offshore wind turbines including aerodynamics, hydrodynamics, and control, in modelling the total system's dynamic surge and pitch motions, and the mooring line tension response of a floating offshore wind turbine, compared to a time-domain method?"*

To answer this question, several sub-questions are formed. The first two are about developing the tool itself, and the last one is about the validation of the tool using time-domain results by applying it to openly available reference designs:

Sub-question 1 How can the linearized control system of a floating wind turbine, including torque and blade-pitch control and a velocity feedback system, be modelled so that all contributions are taken into account in the frequency-domain method?

Sub-question 2 What are representative environmental conditions that can be used to investigate the effectiveness of the aerodynamic, control and hydrodynamic response modelling method adapted in the developed coupled dynamic frequency-domain tool?

Sub-question 3 How effective is the alternative method in linearizing the mooring line dynamics if a nonlinear time-domain method is used for comparison?

Sub-question 4 How effective is the dynamic frequency-domain method compared to a time-domain method in estimating the total system's surge and pitch motions, and mooring line tension for the UMain VolturnUS semi-submersible in combination with IEA's 15 MW reference turbine?

4.2. Research objectives

In order to be able to answer the formed research questions, the following main objective for this thesis is formed:

> *To enhance on RAFT in order to develop an improved, coupled frequency domain model for floating offshore wind turbines including aerodynamics, hydrodynamics and control, so that the model is capable of accurately predicting the system's motions and mooring line tensions.*

Sub-objectives are set up that build up to the main objective. Note that these objectives have a lot in common with the research questions. The first two are about developing the tool, whereas others are about creating deliverables with the tool in order to validate it. The formed sub-objectives are:

Sub-objective 1 Couple RAFT's aerodynamic and control terms with the better potential flow based hydrodynamic model. Therefore the following has to be understood:
(a) What kind of output-files RAFT generates, and if needed, how this file-type can be transformed into a file-type that the better potential flow based hydrodynamic model can read.
(b) How to implement additional inertia, drag and stiffness terms into the better potential flow based hydrodynamic model.

Sub-objective 2 Identify representative environmental conditions that can be used for validation of the designed dynamic frequency-domain tool, so that the effectiveness of the adapted modelling method for the aerodynamics, control and hydrodynamics can be examined.

Sub-objective 3 Implement an improved mooring modelling method to simulate the mooring system's dynamic behaviour in the frequency-domain.

Sub-objective 4 Determine how large the relative contribution of the floater, mooring system, and RNA to the total inertial, drag, spring and excitation forces are, at chosen environmental conditions, for the UMain's VolturnUS semi-submersible in combination with IEA's 15 MW reference turbine.

Sub-objective 5 Compare the surge, pitch and mooring loads power spectral densities for the chosen representative environmental conditions, both obtained via the frequency-domain tool as by the use of time-domain software, for the UMain's VolturnUS semi-submersible in combination with IEA's 15 MW reference turbine.

4.3. Approach

To execute the plan, several software programs will be used. Since MARIN has its own in-house software for modelling the hydrodynamics, including mooring systems, it is chosen to implement the aerodynamics derived from RAFT in these programs. In section's 3.1.4 figure 3.2 it is shown that the aerodynamics are determined using a separate Rotor class. These results are used as input in RAFT's FOWT class, indicating that the input can also be implemented in other software. If this implementation succeeds, MARIN is experienced in evaluating the designs with its own, highly validated software packages. All the in-house developed software have been validated against higher-order fidelity software and high quality model test results. The software used in this thesis are briefly explained in this section, so that the reader knows what kind of software is applied.

DIFFRAC

DIFFRAC can be used for modelling hydrodynamics. The linear potential code is a wave diffraction

program that is capable of calculating first-order wave loads, and mean and low-frequency second-order drift loads for free floating or moored structures using the method as presented in section 3.2.2, as well as the frequency dependent hydrodynamic coefficients. It is applicable for all water depths, i.e. shallow, intermediate, and deep water. Furthermore it can calculate the structure's motion responses, including hydrodynamic interaction if it concerns multiple structures [76].

However, DIFFRAC does not include any form of viscous damping. The damping calculated by the software is purely of potential origin, corresponding to the wave making damping. Any additional linear damping has to be determined and added by the user.

aNyMOOR

For mooring analysis the software aNyMOOR can be used. The software is capable of determining the static mooring line stiffnesses based on the mooring system's input, such as the number of mooring lines, the unstretched mooring line length, anchor and fairlead locations, and the mooring line's material properties. It can do so for multiple type of mooring configurations, amongst others for catenary systems [77].

aNySIM XMF(-fredom)

aNySIM XMF is mainly being used as a time-domain software that simulates the motions of both stationary and sailing vessels. It is based on a time-step solution of the system of coupled differential equations of motion. Vessel specific results from linear diffraction calculations, such as the frequency dependent wave forces, added mass and damping are typically used to model the hydrodynamics, as shown in equation 3.6. These values can be derived from e.g. DIFFRAC. ANySIM can compute the motions of these vessels resulting from non-linear hydrodynamic and mechanical loading. For every time step, it calculates the forces from all the different components acting on the bodies, from which the accelerations follow. It integrates over the time-steps, simulating the motions of the bodies [75].

Besides the time-domain application, it has a build-in frequency-domain functionality. Using the imported frequency dependent databases it solves the system's response by solving equation 3.7, thus in stead of stepping through time, the response is calculated over a range of frequencies.

The advantage of this solver over the solving capability in DIFFRAC is that the mooring stiffness can be linearized in the software itself, instead of the necessity of manually updating the stiffness used in DIFFRAC for each situation yourself. Furthermore, it has capabilities to deal with quadratic viscous damping, which DIFFRAC does not have. Nevertheless DIFFRAC, or another wave diffraction program, is needed to determine the frequency dependent hydrodynamic excitation and coefficients.

5 Methodology

How the software mentioned in section 4.3 will be used in order to improve RAFT will be explained in this chapter. Before all the steps will be explained in more detail, the overall workflow adopted in this work is presented in figure 5.1. Everything inside this figure is included in a Python wrapper. This ensures that if the workflow transitions from one software type to another, the necessary data from the initial software is processed into an appropriate input file format for the subsequent software. As mentioned in chapter 4, this thesis tests the method for UMain's VolturnUS semi-submersible design in combination with IEA's 15 MW reference turbine. All the mentioned parameters of this design are retrieved from the technical report written by Allen et al. [3], unless explicitly mentioned differently.

5.1. Generating the hydrodynamic coefficients

The workflow starts with creating a 3D-model of the floater and prescribe the geometry in a format that is understood by the BEM-software DIFFRAC. To do so, the whole geometry is designed in Rhinoceros 6, and using the Grasshopper extension, the vtk-file is generated. The vtk abbreviation comes from Visualization Toolkit and it is used to store 3D graphic information in a text-based file, which also can be visualized. In this case, it contains the mesh of the geometry that will be used for the BEM calculations. An example of a visualized vtk-file is shown in figure 3.3, showing the mesh used in this study. Note that for linear potential codes, only the submerged part of the geometry is represented.

Besides the vtk-file, general design specific properties must be specified as well. These can be found in technical reports, or, if it is a new design, they can be calculated using analytical calculations. Since the design evaluated in this thesis has a technical report, the values are retrieved from there. One of the inputs required are the radii of gyration, k_{ii}. Since the technical report for UMain's design only gave the moment of inertia about the respective axes, I_{ii}, the radii of gyration are determined using equations 5.1, where m equals the total mass of the design. All the other required parameters are listed in table 5.1, with the hydrostatic matrix given in table 5.2. In Appendix A the VolturnUS-S design plan is given.

Lastly, DIFFRAC also requires environmental data as input. Since the water depth influences the system's response, the water depth needs to be specified. The VolturnUS-S design is designed for a water depth 200 meters. The water density is assumed to be 1025 kilograms per cubic meter in this thesis.

After creating the vtk-file and implementing all the required parameters, DIFFRAC can perform its BEM calculations. This is the most computational expensive step of the entire workflow.

$$\begin{aligned} k_{xx} &= \sqrt{\frac{I_{xx}}{m}} \\ k_{yy} &= \sqrt{\frac{I_{yy}}{m}} \\ k_{zz} &= \sqrt{\frac{I_{zz}}{m}} \end{aligned} \tag{5.1}$$

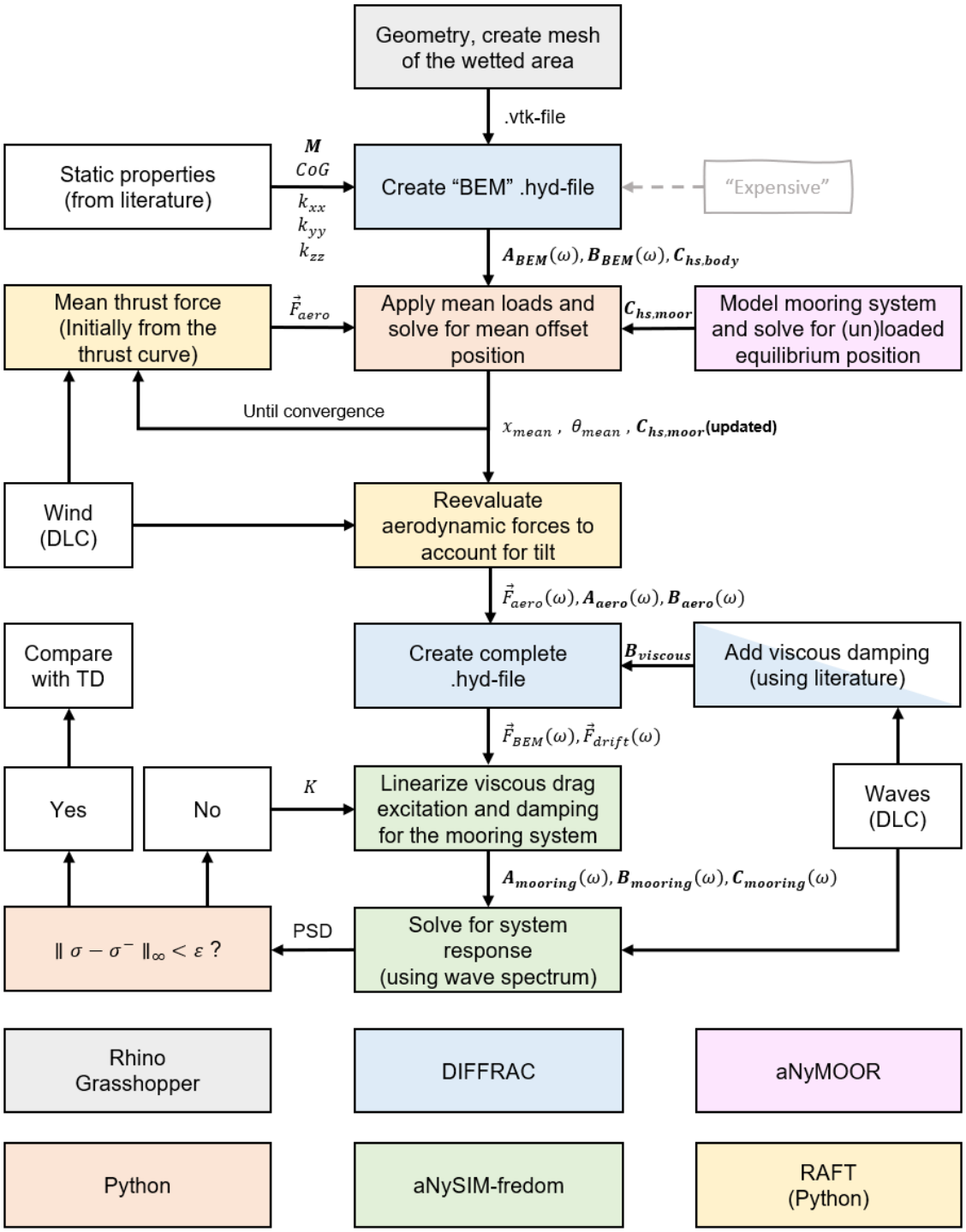

Figure 5.1: Workflow used to couple the different software

Table 5.1: General design specific properties required as input for DIFFRAC calculations, and the values used for the VolturnUS-S design

Parameter	Value	Unit
Length between perpendiculars	90.13	[m]
Mean draft	20	[m]
Beam	102.13	[m]
$COG_{x,y,z}$ w.r.t. keel	[0.32, 0.00, 17.71]	[m]
Mass	20038.803	[t]
Roll radius of gyration	45.34	[m]
Pitch radius of gyration	45.37	[m]
Yaw radius of gyration	34.08	[m]

Table 5.2: VolturnUS-S's hydrostatic stiffness matrix ([N/m], [N/rad] or [Nm/rad])

	Surge	Sway	Heave	Roll	Pitch	Yaw
Surge	0	0	0	0	0	0
Sway	0	0	0	0	0	0
Heave	0	0	4.470e+06	0	0	0
Roll	0	0	0	2.190e+09	0	0
Pitch	0	0	0	0	2.190e+09	0
Yaw	0	0	0	0	0	0

5.2. Estimation of the mean offset position

To estimate the mean offset position, the mean forces acting on the system and the forces counteracting these mean excitation forces have to be determined. The mean offset is found where these are in equilibrium. The excitation forces are design load case (DLC) dependent, making this step DLC dependent. The DLC conditions that apply are listen in a spreadsheet. First, the methodology to determine the counteracting forces is described.

5.2.1. Determining the total stiffness

As mentioned in section 2.2, part of the system's stiffness comes from the floater's hydrostatic stiffness, but a mooring system is needed in order to prevent the floater from drifting away due to external forces acting on the system. Although RAFT's quasi-static mooring modelling approach is defined as a potential improvement in section 3.3.3, the approach was found to be sufficient for initial stage optimization in section 3.2.2. Therefore, such a method is also adopted for making the first estimation of the system's mooring stiffness. In this thesis, the mooring lines for the quasi-static model are modelled in aNyMOOR. In this program the mooring line curves are determined by defining the material properties, as well as the fairlead positions and line orientations.

The material properties for the VolturnUS-S semi-submersible floater design are presented in table 5.3. The design has three catenary lines of an unstretched length of 850 meters, each connected at the fairlead to one of the platform's three outer columns, at a depth of 14 meters below the water line. An overview of the mooring line placements is presented in table 5.4, depicted in appendix A. Note that the origin of the given coordinates is in the center of the floater, at the keel.

Table 5.3: Mooring line properties for the VolturnUS-S semi-submersible [3]

Parameter	Value	Unit
Unstretched length	850	[m]
Anchor depth	200	[m]
Chain diameter	185	[mm]
Break load	22286	[kN]
Mass	685	[kg/m]
Submerged weight	5845	[N/m]
Stiffness EA	3270	[MN]
Fairlead pretension	2437	[kN]

Table 5.4: Mooring line placement and orientation for the VolturnUS-S semi-submersible [3]

Fairlead number [-]	x [m]	y [m]	z [m]	Line orientation [deg]
1	58	0	6	0
2	-29	50.23	6	120
3	-29	-50.23	6	240

Using the data given in these tables, the mooring system characteristics over a range from minus 30 meters, to unloaded conditions, up to a surge offset of 30 meters, as shown in figure 5.2 are found. The figure shows the restoring force in surge direction and moment in pitch direction for the mooring

system over the range of surge offsets. Note how the derivative of the curves alter over the offset, showing the nonlinearity of the mooring system.

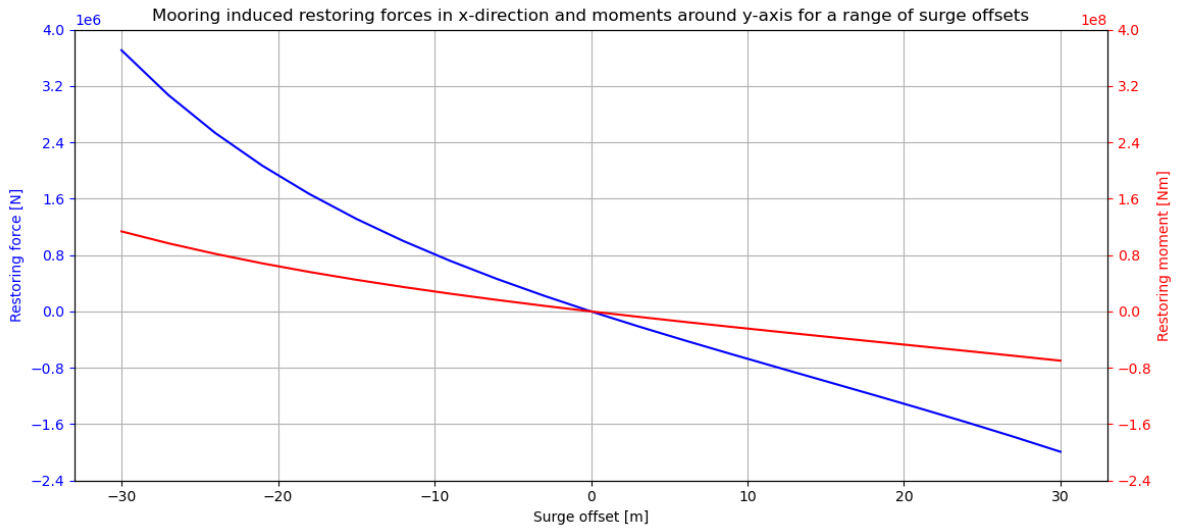

Figure 5.2: Mooring induced restoring forces in x-direction and moments around y-axis for a range of surge offsets

For these curves, the floater is assumed to remain in the upright position. If the floater is pitching, the restoring forces also alter. Therefore the mooring system characteristics for a range of pitching angles is also determined using aNyMOOR. However, since the tensions in the lines differ over the surge offset, this is done for every five meters of offset. In this thesis the wind and waves are assumed to come from positive x-direction, the curves are only shown for negative surge offsets. Figure 5.3 shows the mooring's restoring force in surge-direction, while the restoring moment in pitching direction is shown in figure 5.4. Note that for the tower in upright position, i.e. zero pitch angle, the values match with the restoring forces found in figure 5.3.

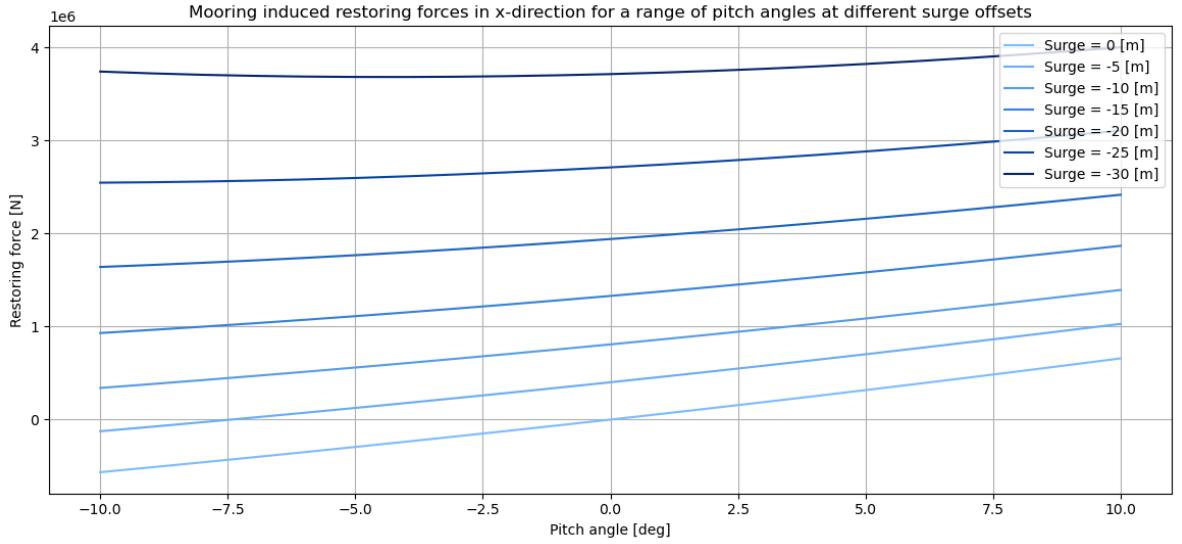

Figure 5.3: Restoring force in surge direction for a range of pitching angles, at different surge offsets

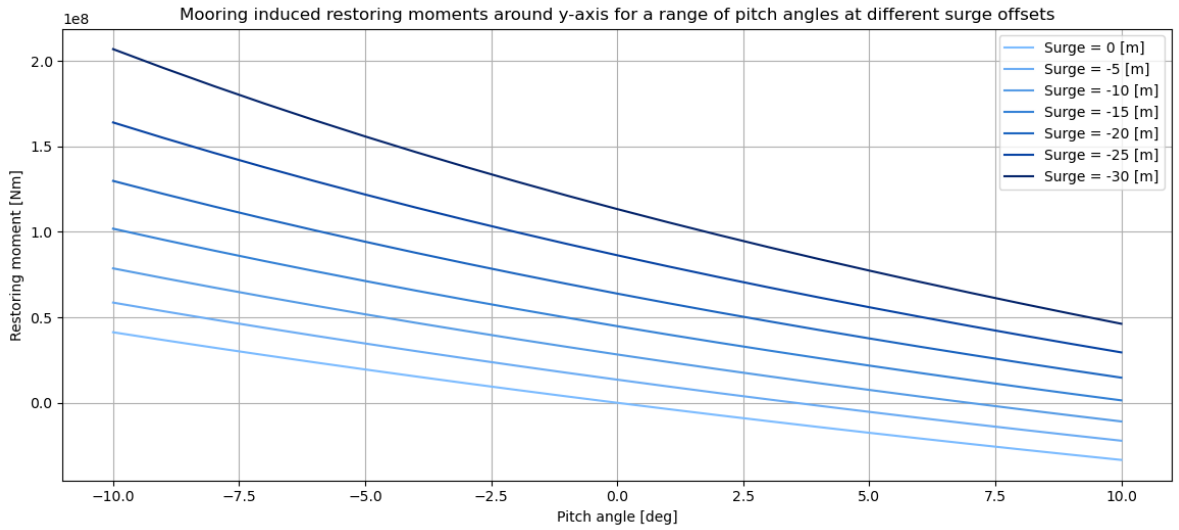

Figure 5.4: Restoring moment in pitching direction for a range of pitching angles, at different surge offsets

As can be seen in the figures, the surge offset influences the restoring forces for different pitching angles, but a pitching angle also influences the restoring force in surge direction. Therefore the offset is determined using an iterative loop to find the actual force equilibrium.

Besides the stiffness created by the mooring system, the floater itself also has its hydrostatic stiffness, which is shown in table 5.2. The total pitching stiffness is the sum of both contributions, as can be seen in equation 3.9. Where the mooring system creates all the stiffness in the surge direction, the hydrostatic stiffness makes up for the most of the stiffness in pitch direction for the VolturnUS-S design. The mooring restoring moment is roughly 1 percent of the total pitching restoring moment for this design. Note however, that this is a semi-submersible design. For e.g. a TLP design the mooring contribution would play a much bigger role, as mentioned in sections 2.1 and 2.2.

5.2.2. Aerodynamic forces

For the initial estimation of the mean thrust force, the thrust curve of the turbine is used. The generated thrust at the DLC wind speed is looked up using this curve, and the moment around the y-axis is calculated by multiplying the thrust with the distance between hub height and the center of gravity in z-direction. The thrust curve of the turbine installed on the VolturnUS-S design is given in figure 5.5. By applying these forces and using the stiffnesses as described in the previous subsection, an initial mean offset is determined.

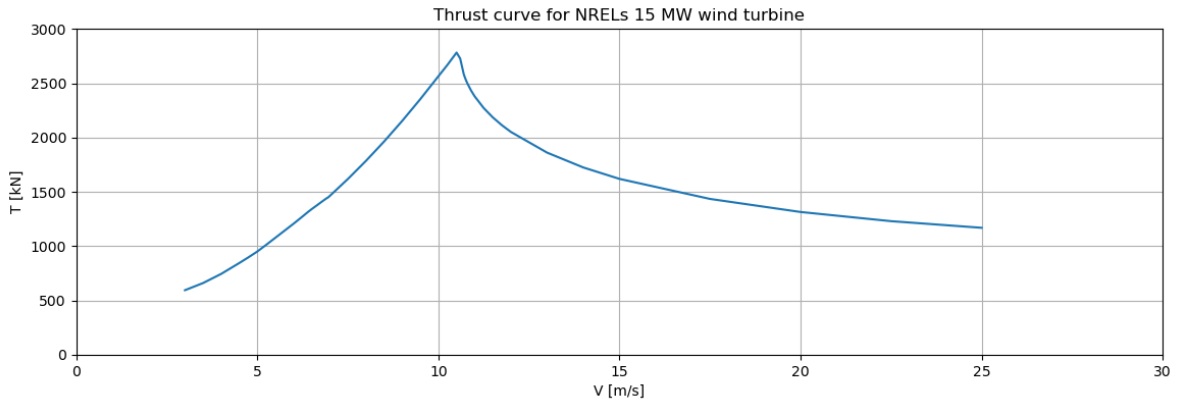

Figure 5.5: The thrust curve for NREL's open-sourced 15 MW wind turbine [54]

Since the turbine is now operating under a platform pitch angle, the actual thrust force differs from the

one determined using the look-up table as the angle of attack on the blades alter. The aerodynamics are therefore re-evaluated using this initial mean offset, i.e. the static pitch angle. This is done using RAFT's rotor class, which in turn uses the BEM* package CCBlade to determine the rotor averaged thrust and torque and their load derivatives for the mean operating conditions, while applying the correction factors to BEM* theory in CCBlade that are mentioned in section 3.2.3. CCBlade requires the turbine specific input, which are listen in a YAML-file as shown in appendix B, and contains the rotor design, airfoil data, and control settings. In addition, the static platform pitch angle is added to this information in Python. This way the torque and blade pitch control settings matching with the mean wind speed are taken into consideration, as well as the mean wind speed itself and the platform pitch angle, so that the angle of attack of the turbine blades is matching the DLC conditions.

Modifications to RAFT's Rotor class

However, as mentioned in section 3.3.3, RAFT does not include the nacelle velocity feedback in this determination. They tried to implement the gain as given in the GitHub of the turbine ([52]), which has seconds as its unit and is applied to the turbine's angular velocity, rather than the horizontal velocity as was used in the linearization process of the aerodynamics as shown in section 3.2.3. Therefore, modifications in RAFT's Rotor class have to be made in order to include the feedback term in the software. As a reminder, the nacelle velocity feedback term was included in the blade pitch control as:

$$\Delta\beta = k_{P\beta}\Delta\Omega + k_{I\beta}\int \Delta\Omega dt + \underbrace{k_{Px}\dot{x}}_{\text{nacelle velocity feedback}}$$

So in order for the gain to give radians as output after the multiplication with the floater's horizontal velocity, the gain has to be converted to have radians seconds per meter as units, in stead of seconds.

Furthermore, RAFT used one gain value for all frequencies, whereas the actual feedback utilizes a second-order low-pass filter around the platform's pitching natural frequency as well as a notch filter around the tower fore-aft natural frequency [3]. This means that the gain value alters over the frequencies.

A second-order low-pass filter attenuates the gain above a certain cut-off frequency, ω_c. The damping ratio, ζ, can be set to determine the response speed and overshoot or oscillation behaviour. If it is set to one, the system is considered as critically damped. That means that the system's response to a step input is the fastest possible without any overshoot. A higher damping ratio results in a slower response, while a lower damping ratio allows for a faster response. If the damping ratio is very low, the filter responds rapidly to input changes, introducing overshoot and oscillations.

To determine the proportional gain for a second-order low-pass filter over a frequency range, the following transfer function can be used:

$$H_{LP}(s) = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n} \tag{5.2}$$

wherein ω_n is the natural frequency related to the cut-off frequency, s is the complex frequency, and ζ is the damping ratio. The damping ratio for NREL's 15 MW turbine is equal to one, making it a critically damped filter, which simplifies the equation. The relation between the cut-off frequency and the natural frequency is as follows:

$$\omega_c = \frac{\omega_n}{\sqrt{2}} \tag{5.3}$$

Substituting this relation in the transfer function and simplifying for the magnitude of the frequency response gives:

$$|H_{LP}(j\omega)| = \frac{1}{\sqrt{1 + \left(\frac{\omega}{\omega_c}\right)^4}} \tag{5.4}$$

The notch filter is typically implemented as a second-order band-reject filter. The center frequency determines the center of the frequency-band and the quality factor determines the width of the notch. A lower quality factor results in a wider notch, and thus the filter shall have a broader attenuation range around the center frequency. For higher quality factors the reasoning works vice versa. The notch

filter transfer function, $|H_N(j\omega)|$, can be programmed in the frequency-domain using Python. The total response is then given as:

$$|H_{tot}(j\omega)| = |H_{LP}(j\omega)| \cdot |H_N(j\omega)| \tag{5.5}$$

To illustrate the working of both transfer functions, figure 5.6 is generated. Note that the second-order low-pass filter is the one mainly determining the gain value, since it attenuates the signal above the cutoff frequency, and therefore the signal is already close to zero when the effect of the notch filter starts attenuating. The notch filter is probably left as is from the non-floating tower application. The gain value itself for each frequency is determined by multiplying the transfer function times the gain as given in equation 5.6. How this methodology is implemented in RAFT's Rotor class is given in appendix C.

$$k_{Px}(\omega) = |H_{tot}(j\omega)| \cdot k_{Px} \tag{5.6}$$

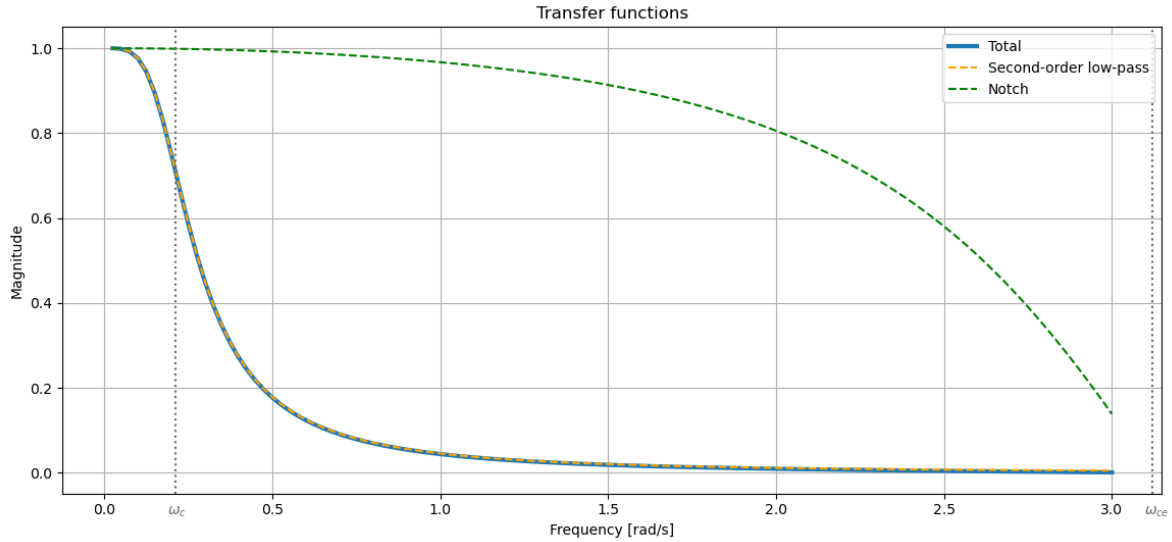

Figure 5.6: Magnitude of the transfer functions for the velocity feedback gain

5.2.3. Equilibrium position

Now the mean excitation forces and contributions responsible for counteracting these forces are determined, the equilibrium position can be calculated. Since the mean thrust force and the mooring stiffness alter for different pitching angles, this position is found using an iterative loop, depicted in figure 5.7, which shows the loop as shown in figure 5.1 in more detail.

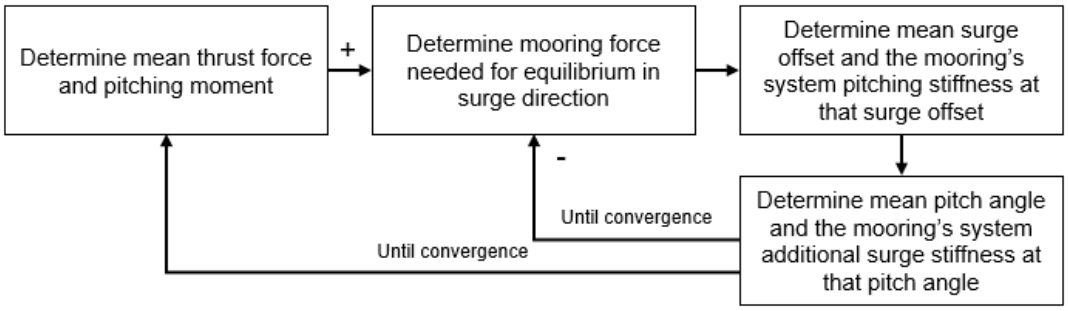

Figure 5.7: Iterative method to determine the mean surge and pitch offsets due to a thrust force compensated by hydrostatic and mooring stiffness

5.3. Generating the total coefficients

Once the mean offset is determined, the final aerodynamic forces can be determined. Using RAFT's rotor class the frequency dependent aerodynamic added mass matrix, damping matrix, and excitation vector are determined using the Python function shown in appendix D.1. These are transformed into the six DOF contributions acting at a desired reference position using the method as described by Sadeghi and Incecik [101]. By taking the same reference position as used in the hydrodynamic calculations from section 5.1, i.e. the center of gravity position, the contributions can be superimposed as is done in equation 3.9, so both contributions are considered in the EOM.

Besides the aerodynamic contributions, the viscous damping of the floating body needs to be added as well in order to accurately simulate the system's response. As mentioned in chapter 3.2.2, the determination of these terms can be expensive. This thesis uses a floater design wherefore these tests have been performed, and therefore uses these results as an input. The given quadratic viscous damping is presented in table 5.5. The quadratic values are linearized based on the DLC conditions, by multiplying the quadratic value with a root mean square (RMS) value of the velocity response. The velocity response is determined using the DLC conditions a DIFFRAC run, wherein the linear viscous damping is assumed to be 3% of the critical damping for each DOF, and the mooring stiffness is based on the previous described quasi-static method. The code can be seen in appendix D.3. Finally, the once DLC dependent hydrodynamic linearized viscous damping matrix is determined, another DIFFRAC run is executed that includes the hydrodynamic coefficients, aerodynamic coefficients, and the hydrodynamic linear viscous damping values, in order to re-evaluate the drift forces including these terms. The function is given in appendix D.4. It generates a hyd-file that will be used as input in aNySIM, containing all the frequency dependent terms. Note that for this DIFFRAC run, the mooring contribution is not included, as these will be determined in aNySIM next.

If one would like to adapt this workflow for a floater of which the viscous damping is not yet determined, one could e.g. use Morison elements for the slender bodies and estimate the viscous drag using the method presented by Housseine, Monroy, and Hauteclocque [51].

Table 5.5: VolturnUS-S's quadratic viscous damping matrix ([Ns^2/m^2], [Ns^2], [Ns^2/m] or [Nms^2])

	Surge	Sway	Heave	Roll	Pitch	Yaw
Surge	9.225e+05	0	0	0	-8.918e+06	0
Sway	0	9.225e+05	0	8.918e+06	0	0
Heave	0	0	2.296e+06	0	0	0
Roll	0	8.918e+06	0	1.676e+10	0	0
Pitch	-8.918e+06	0	0	0	1.676e+10	0
Yaw	0	0	0	0	0	4.798e+10

5.4. Solving for the system response

All the contributions in equation 3.9 are included in the hyd-file, except the mooring system dependent one. To try to improve the prediction of the mooring tension response, the mooring system is modelled in aNySIM. The generated hyd-file is used as input for aNySIM. In the frequency-domain solver of aNySIM, the mooring stiffness is linearized based on an initial offset, and a position and velocity perturbation step from that offset position. The mean offset position is based on the quasi-static estimation, as previously elaborated on in section 3.22. Since the quasi-static mooring method mainly neglects the drag effects that become significant at higher platform motion frequencies, the velocity perturbation step is determined so that it takes the drag better into consideration. To do so, the method as suggested in Housseine, Monroy, and Hauteclocque [51] is adapted.

The method starts with an initial guess of the velocity perturbation step, that is used to determine the response using that step size. It assumes that the wave velocity for irregular waves can be described as a Gaussian distribution with a mean of zero. Thereby, the linearization coefficient K is obtained by minimizing the averaged least square error between the linear and the non-linear drag load, as in [119]:

$$K = \frac{< |u|u^2 >}{< u^2 >} = \frac{< |u|^3 >}{\sigma^2} = \sqrt{\frac{8}{\pi}}\sigma \tag{5.7}$$

where values denoted between $<>$ are the expected values and σ is the wave velocity standard devia-

tion. Convergence is achieved when the difference between the current standard deviations vector, σ, and the previous standard deviations vector, σ^-, is below a certain threshold ϵ. In this work, a threshold of 1 percent is used.

How this process is adapted in this thesis is illustrated in figure 5.8 in more detail. Note that $||.||_\infty$ is the infinite norm, meaning that the maximum absolute value for all DOF is used as criteria. Convergence is only achieved when all DOF are within the divergence criteria.

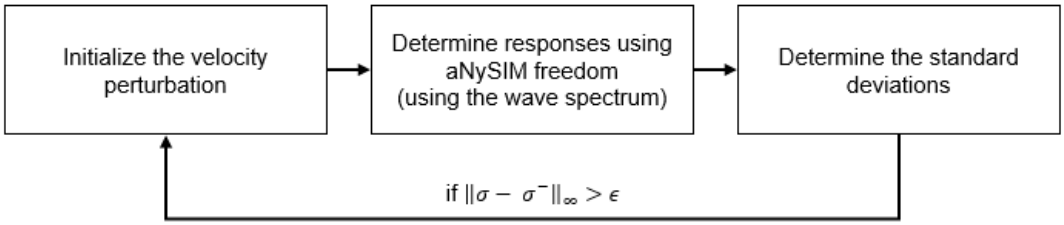

Figure 5.8: Mooring linearization process

If convergence is achieved, the results are final for the whole simulation. All the terms from equation 3.9 are determined during the process, and aNySIM fredom determines the motions using:

$$\vec{x}(\omega) = \frac{\vec{F}(\omega)}{-\omega^2 \left(\mathbf{M} + \mathbf{A}(\omega)\right) + i\omega\mathbf{B}(\omega) + \mathbf{C}(\omega)} \tag{5.8}$$

6
Results

This chapter shows the results that were found by applying the methodology as explained in chapter 5. Before it shows the results, some intermediate steps will be evaluated to show whether the software does what it is supposed to do at that point, the verification. After these verification checks, the model will be compared with time-domain results for chosen design load cases using the VolturnUS-S design, the validation. Based on these results the recommendations for follow on work will be drafted.

6.1. Verification

The workflow presented in figure 5.1 applies different software at different moments throughout the process. Most of the software used are MARIN in-house developed tools. As mentioned in section 4.3, these software are highly validated against other higher-fidelity software tools and high quality model tests. Therefore, only verification of the newly adapted approach will be checked for verification. I.e., the aerodynamic contributions determined with RAFT, with and without the nacelle velocity feedback term, and the implementation of these terms in the MARIN software. The next section starts with the aerodynamic contributions verification.

6.1.1. Aerodynamic contributions

RAFT's rotor class is used for the determination of the aerodynamic contributions. To test whether the aerodynamic contributions calculations implemented in this model give the same results as shown in figure 3.8, the same controller settings, i.e without the velocity feedback gain active, turbulence levels, and platform reference point were used in a simulation with the new workflow. One wind speed below, one at, and one above rated wind speed is chosen for illustration purposes, so that all different control strategies are demonstrated, i.e. torque and blade-pitch controlled instances. The results are shown in figure 6.1 and show the same results as the ones from RAFT, giving confidence in the implementation of the class in the model.

To see how the changes made in the Rotor class regarding the nacelle velocity feedback contribution influence the outcome of the aerodynamic contributions, the same exercise is done for both with and without the nacelle velocity feedback active. This time, the reference point is at COG in stead of at sea water line level, so that a direct comparison with the hydrodynamic added mass and damping determined by the BEM software can be made. RAFT determines the initial frequency dependent values for the fore-aft motion at hub height, i.e. in surge direction. Because of the rigid-body assumption applied in this thesis, these are therefore equal to the "$_{11}$" motion. This makes that the hydrodynamic a_{11} and b_{11} can be used for the first comparison. The results of this exercise are shown in figure 6.2. The dotted lines in the figures represent the results without the velocity feedback active, and the solid lines with. Note that only the frequency dependent added mass and damping are presented in this figure, and not the excitation coefficients and rotor averaged wind speed amplitudes. The latter remain unaltered due to a change in the velocity feedback control settings, as can be seen in equation 3.41.

The aerodynamic added mass remains lower than the hydrodynamic added mass over the whole frequency range. The highest aerodynamic added mass is experienced around 0.02 Hertz, and at that point it is roughly 40 percent of the floater's weight. For the damping it holds that the aerodynamic

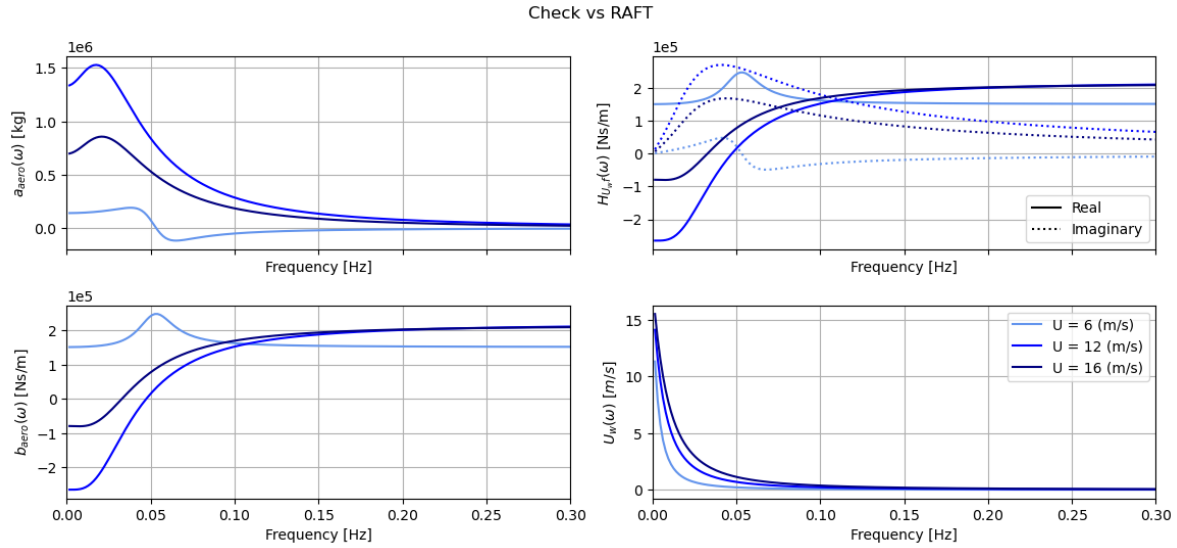

Figure 6.1: The calculated aerodynamic added mass, damping, and excitation coefficients and rotor averaged wind speed amplitudes for the IEA 15-MW Reference turbine over three wind speeds, one below, one around, and one above rated wind speed, to compare the results with RAFT results

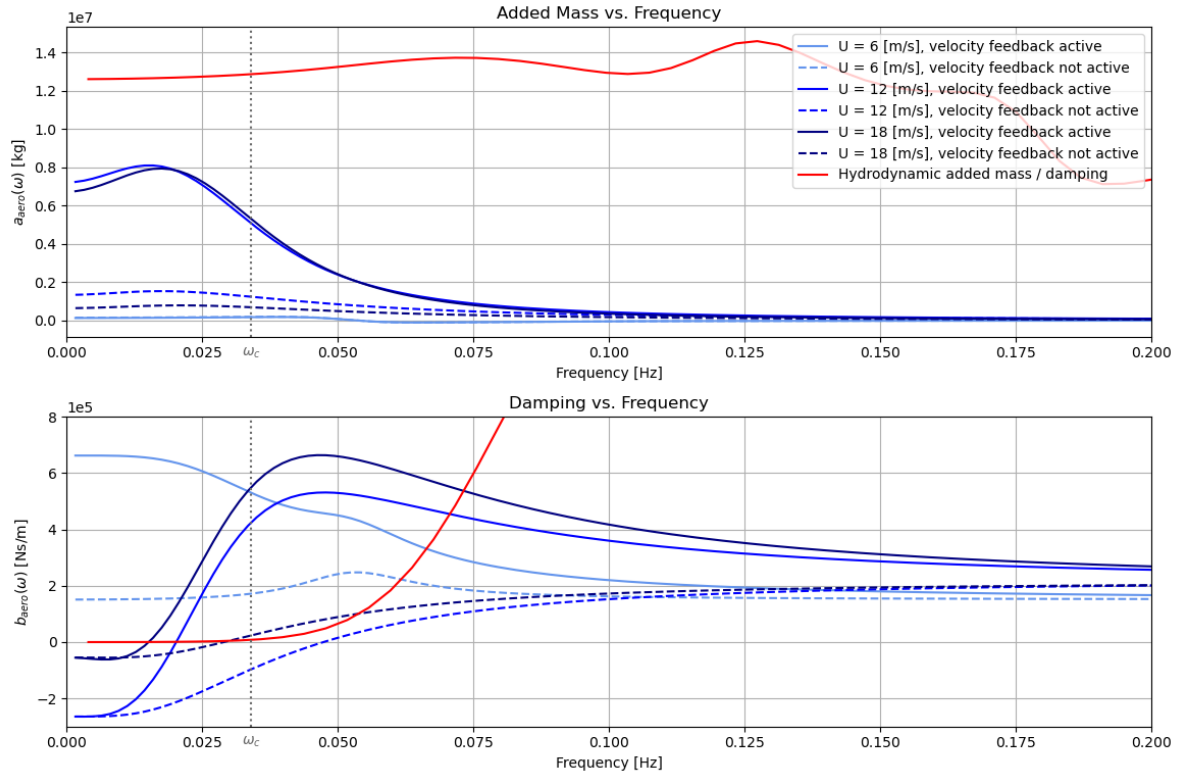

Figure 6.2: The calculated aerodynamic added mass and damping for the IEA 15-MW Reference Turbine over three wind speeds, one below, one around, and one above rated wind speed, with and without the adjusted control settings for the velocity feedback, as well as the hydrodynamic coefficients

damping is the most important in the lower frequency range, in the higher frequency range the floater's damping is dominant, with a peak value of 8E6 at roughly 0.2 Hertz.

The differences between the responses with and without the adjusted control settings alter for the wind speeds below rated wind speed compared to the above rated wind speeds. To understand why

this happens, equation 3.41 is rewritten to contain only the below and above rated wind speed control settings.

In the below rated regime, the turbine is torque controlled, meaning that $k_P\beta$ and $k_I\beta$ are zero. This results in the following equations for the aerodynamic added mass and damping:

$$\begin{aligned} a_{aero_{below}}(\omega) &= \mathfrak{Re}\left[\frac{1}{i\omega}[T_U - T_\beta k_{Px} - \frac{T_\Omega i\omega(Q_U - Q_\beta k_{Px})}{I_r\omega^2 + (Q_\Omega - N_g k_{P\tau})i\omega - N_g k_{I\tau}}\right] \\ b_{aero_{below}}(\omega) &= \mathfrak{Re}\left[T_U - T_\beta k_{Px} - \frac{T_\Omega i\omega(Q_U - Q_\beta k_{Px})}{I_r\omega^2 + (Q_\Omega - N_g k_{P\tau})i\omega - N_g k_{I\tau}}\right] \end{aligned} \tag{6.1}$$

For the wind speeds above rated wind speed, the turbine is blade-pitch controlled. In that case, $k_{P\tau}$ and $k_{I\tau}$ are zero, resulting in:

$$\begin{aligned} a_{aero_{above}}(\omega) &= \mathfrak{Re}\left[\frac{1}{i\omega}[T_U - T_\beta k_{Px} - \frac{(T_\beta k_{I\beta} + (T_\Omega + T_\beta k_{P\beta})i\omega)(Q_U - Q_\beta k_{Px})}{I_r\omega^2 + (Q_\Omega - Q_\beta k_{P\beta})i\omega + Q_\beta k_{I\beta}}\right] \\ b_{aero_{above}}(\omega) &= \mathfrak{Re}\left[T_U - T_\beta k_{Px} - \frac{(T_\beta k_{I\beta} + (T_\Omega + T_\beta k_{P\beta})i\omega)(Q_U - Q_\beta k_{Px})}{I_r\omega^2 + (Q_\Omega - Q_\beta k_{P\beta})i\omega + Q_\beta k_{I\beta}}\right] \end{aligned} \tag{6.2}$$

The dotted lines in figure 6.2 are for the case where there is no velocity feedback control, meaning that in the equations k_{Px} is equal to zero. The differences therefore come from the terms that are multiplied with k_{Px}.

In the lower frequencies we see that for the below rated region the added mass is almost unaltered, but the damping is significantly increased over this range. This is due to the fact that without the feedback gain, the blade pitch is not controlled for this wind speed, but now it is. In the above rated region the blade pitch is already being controlled, so the starting point remains the same for the damping, but due to increased pitch control the damping increases immediately afterwards. The feedback method also influences the added mass, which becomes much higher due to the additional terms the velocity feedback term introduce in equation 6.2.

For both above as below rated regions it counts that the biggest effects are seen in the lower frequency range, which is logical since the gain is attenuated after the platform's pitching natural period, as was shown in figure 5.6.

If the aerodynamic added mass and damping are being transformed to be expressed in the six-by-six matrix, the outcome is that there are only contribution in the (coupled) surge and pitch contributions. This meets the expectations since the aerodynamic contributions in this instance are calculated for wind speeds aligned with the x-axis, therefore only these terms are to be expected. Figures 6.3 and 6.4 show the aerodynamic (coupled) surge and pitch contributions for three wind speeds, as well as their hydrodynamic contributions.

Between a_{11} and a_{15} is a factor equal to the arm between hub height and the COG, which is again true for between a_{15} and a_{55}. The same goes for the damping matrix. It becomes clear that the hydrodynamic contributions in surge dominate, but (coupled) with pitch motions the aerodynamic contributions become more pronounced, especially in the lower frequencies. Again this is a result of the nacelle velocity feedback being active in that frequency range. In figure 6.2 it can be seen that all the aerodynamic contributions without the nacelle velocity feedback active are of comparable values as the below rated value, i.e. the light blue line in the figures, for the added mass, and that the damping of the above rated cases remain beneath the below rated, and even negative for a much longer frequency range. It is to be expected that the turbine's relative contribution becomes larger for pitching motions due to the hub height. A surge motion in the COG has the same motion at hub height, whereas a pitching motion at COG induces a large fore-aft motion at hub height which influences the aerodynamics as shown in section 3.2.3.

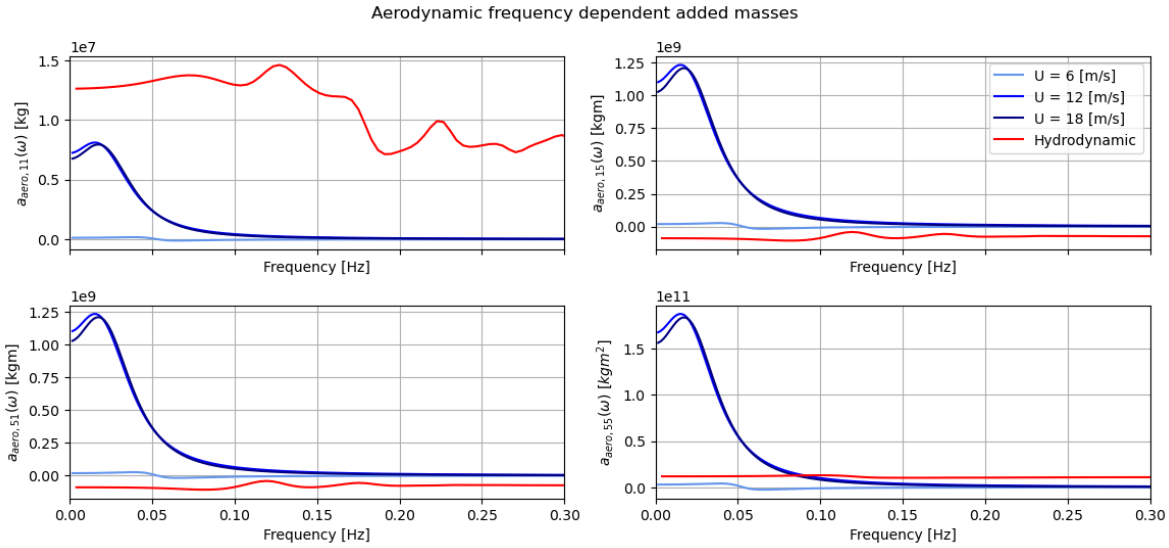

Figure 6.3: Aerodynamic added mass (coupled) surge and pitch contributions for the IEA 15-MW Reference Turbine over three wind speeds, one below, one around, and one above rated wind speed, calculated with respect to the overall COG

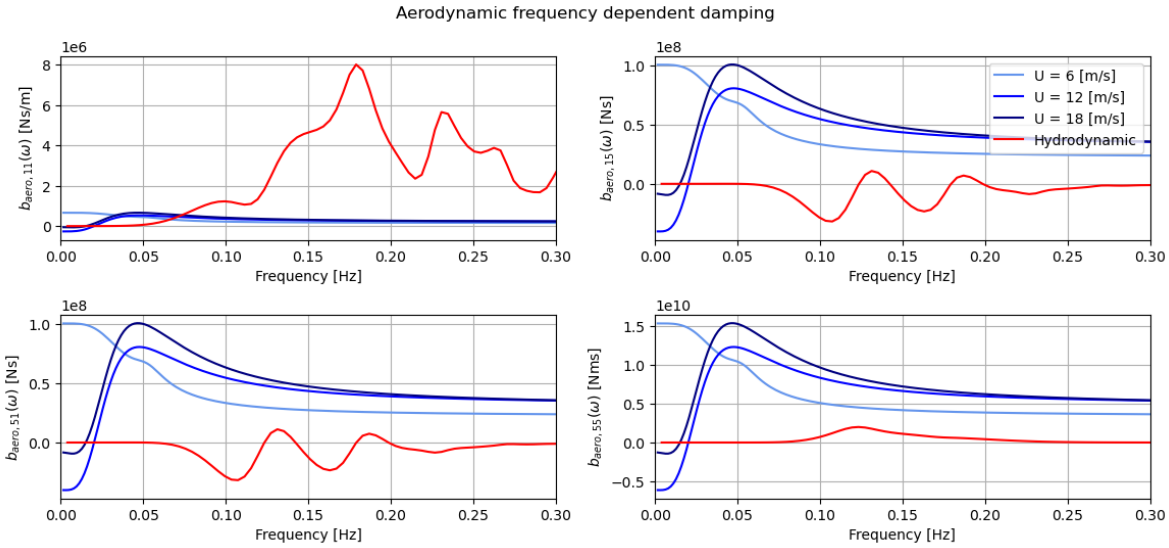

Figure 6.4: Aerodynamic damping (coupled) surge and pitch contributions for the IEA 15-MW Reference Turbine over three wind speeds, one below, one around, and one above rated wind speed, calculated with respect to the overall COG

If the aerodynamic excitation is being recalculated to be around the right reference point and being expressed in all the 6 degrees of freedom, only a force in the x-direction and a moment around the y-axis is found. This matches with the expectations because as mention in section 5.2.1, for now the aerodynamic forces are only calculated for wind speeds aligned with the x-axis. The frequency dependent aerodynamic force in the x-direction and the moment around the y-axis are plotted in figure 6.5. The development of the two plots over the frequencies are equal, but there is a factor equal to the hub height towards COG between them.

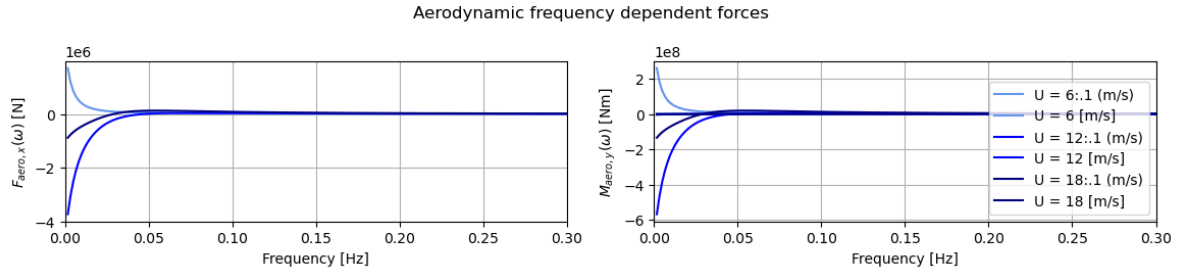

Figure 6.5: The calculated aerodynamic excitation for the IEA 15-MW Reference Turbine over three wind speeds, one below, one around, and one above rated wind speed.

Note that these are only the frequency dependent aerodynamic contributions. RAFT also calculates the mean loads due to the turbine operating at a certain wind speed. These mean loads at the reference point are presented in table 6.1.

Table 6.1: Mean aerodynamic loads for the IEA 15-MW Reference Turbine over three wind speeds, one below, one around, and one above rated wind speed.

Hub height wind speed [m/s]	F_x [N]	F_y [N]	F_z [N]	M_x [Nm]	M_y [Nm]	M_z [Nm]
6	7.98e+05	9.13e+02	-6.02e+02	3.26e+05	1.28e+08	4.90e+05
12	1.77e+06	3.71e+03	-2.28e+03	7.46e+05	2.90e+08	7.44e+05
18	1.03e+06	1.20e+04	-1.36e+04	1.14e+05	1.77e+08	-2.17e+06

If these values are being compared with the ones from before the translation towards the reference point, the translations are being performed correctly. The forces remain the same due to the rigid body assumption, and the moments are the moment at hub height plus the relevant force times the arm between the hub height and COG.

Overall the results found by applying the adjusted RAFT Rotor class make sense. Testing the same conditions as was done in the paper by Hall et al. [46] gives the same results, implying the Rotor class is modelled correctly within the built Python wrapper. Using the adjusted velocity feedback modelling gives higher aerodynamic contributions in the lower frequency range, which is where the feedback control is active, and in the higher frequency range it converges back towards the values without the gain, since there is no feedback term active anymore in that range. The contributions in pitching motions are much larger due to the hub height, and this hub height also explains the characteristics between the forces and moments acting at hub height and acting at COG.

6.1.2. Implementation of the aerodynamic contributions in MARIN software

Now confidence is gained in the application of RAFT's adjusted Rotor class, the last verification that needs to be performed is to check whether these contributions are considered correctly in the software that includes the terms. To do so, the RAOs of the system are evaluated in aNySIM, which uses the hyd-file containing all the frequency dependent contributions as mentioned in section 5.3. The RAO is also determined for the case where there are no aerodynamic contributions written onto the hyd-file, so it can be seen whether it makes a difference or not. Since the main interest is for waves without an incoming angle, the RAOs for the surge and pitching are shown in figure 6.6. Since the orientation of the floating offshore wind turbine was defined symmetrically about the x-axis the forces in sway, roll and yaw are symmetric about the surge-heave plane yielding a net force or moment of zero, when considering a wave coming from the x-direction. This effectively creates a RAO of zero at all wave frequencies for the DOFs of sway, roll, and yaw in this wave heading [3].

The hydrodynamic RAOs shapes match with the ones given in the techreport of the VolturnUS-S design [3]. As can be seen, the additional added mass and damping influence the RAOs magnitudes, indicating that the method of writing them on the hyd-file is working. The changes in RAO can be explained using the added masses and damping as given in figures 6.3 and 6.4. In surge direction, the

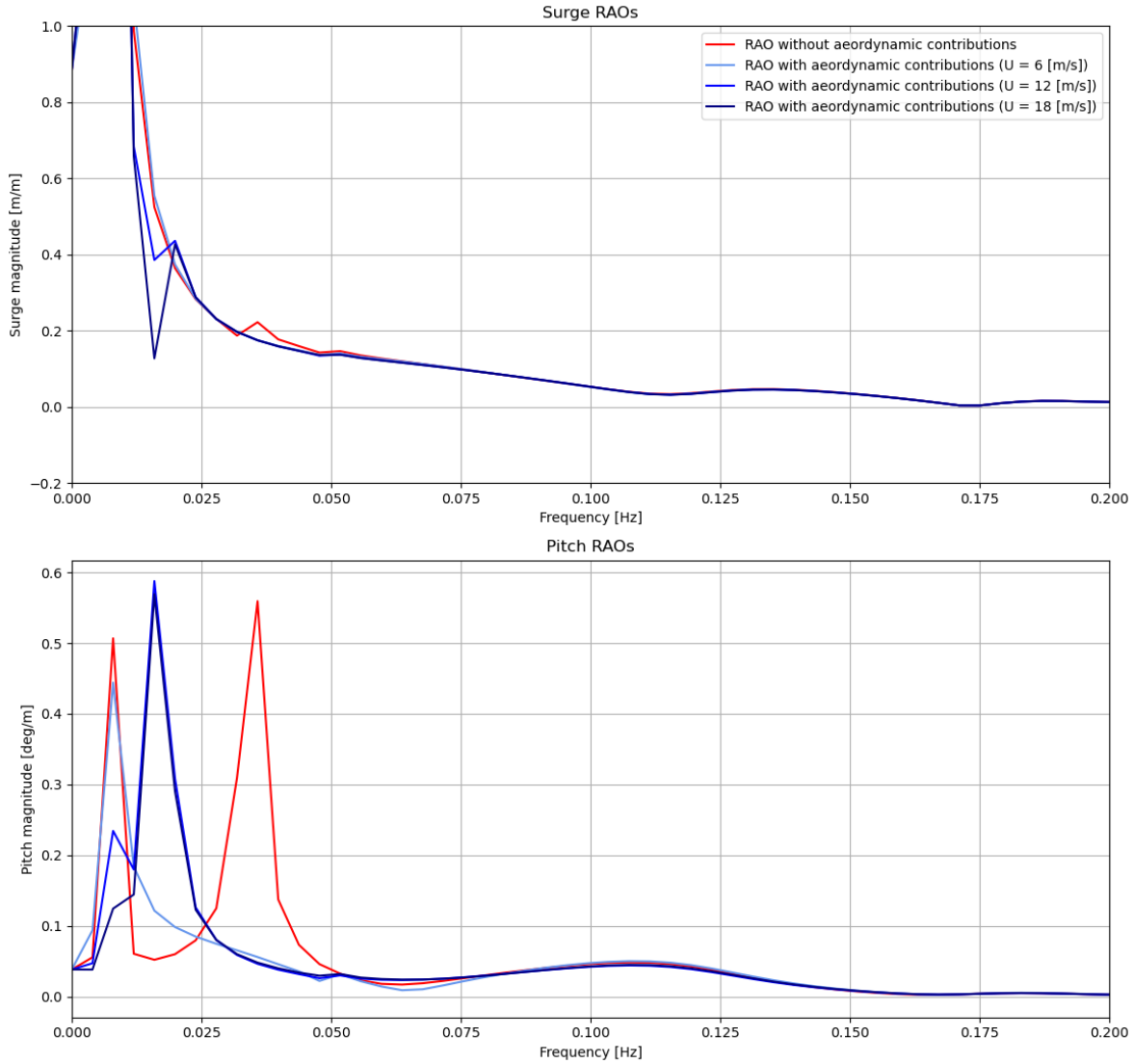

Figure 6.6: Surge and pitch RAOs as determined by aNySIM fredom, for the case without aerodynamic contributions, and for the three cases with aerodynamic contributions.

added mass and damping are relatively small compared to the (added) mass and damping the floater itself experiences already. The additions do dampen the RAO slightly, but not much. In pitch however, the contributions are relatively larger as shown in figure 6.4. As a result, the pitch RAO is altered more due to the addition of the aerodynamic terms. They shift the pitch natural frequency for the above rated wind speeds towards the peak of the added masses, since the natural frequency is a function of the ratio of the system's stiffness and mass, a higher (added) mass leads to a lower natural frequency. These behaviours thus match with the expected behaviour, indicating that the software processes the additional aerodynamic contributions in a correct manner. Therefore the python wrapper is considered verified to a satisfactory extent by the author.

6.2. Validation

Now the Python wrapper applying the methodology as shown in figure 5.1 is considered verified to a satisfactory extent, it can be tested for various design loading conditions (DLCs). This will be done in this section, by first briefly explaining the DLCs that will be tested, followed by the actual results. These results are compared with higher-fidelity time-domain results. Based on the outcomes of both the FD results as the TD results, some recommendations for future work are formed.

6.2.1. Design load cases

The first DLC that will be tested is the same one as used in the paper by Hall et al. [46], wherein RAFT is tested with the results from figures 3.10 and 3.11 as a result. Comparing the new methodology with those results will show whether the method improved on RAFT or not. The other DLCs are based on the VolturnUS-S reference design its technical report [3]. There are many DLCs given in that report that are based on meta data. For classification rules, the FOWT needs to be tested over a large range of wind speeds, with matching wave data for those wind speeds. Many of the load cases are considering extreme events, such as a sudden shut down. For these kind of events TD simulations are needed as explained in section 3.1.2. However, DLC condition 1.1 is about the normal operating conditions, which is wherein the FOWT will be operating most of its time, and is a condition that can be evaluated using FD simulations. In stead of testing for the whole wind speed range as required by the classification societies, three wind speeds are chosen to evaluate the model with. Just as in section 6.1, it is chosen to test for one wind speed below, one around, and one above rated wind speed. These conditions come with their own matching wave spectrum. By choosing these wind speeds the whole range of control methods is examined. The DLCs that will be tested are presented in table 6.2.

Table 6.2: DLCs used for validation

DLC [-]	Wave spectrum [-]	Peak period [s]	Significant wave height [m]	Shape factor [-]	Wave heading [deg]	Wind condition [-]	Hub height wind speed [m/s]	Wind heading [deg]
1	JONSWAP	12.00	6.00	1.00	0.00	Steady	8.00	0.00
2	JONSWAP	8.31	1.18	1.00	0.00	Steady	6.00	0.00
3	JONSWAP	7.44	1.84	1.00	0.00	Steady	12.00	0.00
4	JONSWAP	8.05	3.06	1.59	0.00	Steady	18.00	0.00

6.2.2. Simulations

Before looking at the results, two announcements have to be made. The first is that the new method is dubbed ”RAFT 2.0”, thus all the responses plotted for ”RAFT 2.0” are determined using the methodology as shown in figure 5.1. The second is that the frequency-domain solver in aNySIM turned out not to be capable of dealing with external frequency-dependent forces. It determines excitation forces solely based on the first and second order wave loads read from the hyd-file in combination with the specified wave spectrum. Therefore, the results are plotted starting from 0.05 Hertz, as the TD results would have a high peak in the low frequency range due to wind excitation in that range, while the FD model now only computes responses in the wave frequency range. By not plotting this low frequency range, a better focus is accomplished on the wave frequency range. In appendix E the response for DLC 1 is given from 0 Hertz to demonstrate the amount of energy that the FD model misses in this range.

The results for the surge, pitch, and mooring tension response for the first DLC are shown in figure E.1. Herein the results of multiple methods are shown: The results from RAFT, matching with those from figures 3.10 and 3.11, the results from the improved method RAFT 2.0, the results from the higher-fidelity TD software aNySIM, converted to FD by applying Welch’s method, and the results for the case where the aerodynamic coefficients are not added into the hyd-file but only the mean aerodynamic forces are applied, the ”Hydro” run. Looking at the results, it becomes clear that by using BEM software the surge response predictions improve, even for the case where the aerodynamic coefficients are not included in the simulation. However, for the pitch response this does not hold anymore. The pitch predictions for the improved method match well with the TD results, but the response without the aerodynamic contributions less so. The explanation for this are the coefficients as shown in figures 6.3 and 6.4, resulting in the RAOs as shown in figure 6.6. There, it became clear that the impact of the aerodynamic coefficients have a bigger impact on the pitch motions compared to the surge motions, which is seen in these responses as well. Nevertheless, the hydro response shows how important the 3D BEM modelling is for the pitch motion predictions, compared to the 2D strip-theory approach applied in RAFT. This confirms the theoretical explanation given in section 3.3.2.

Regarding the mooring line tensions, it turned out that the new mooring modelling did not improve the predictions, in fact, the predictions became worse compared to RAFT. The mooring linearization process does ensure that the frequency peaks and frequency range for the mooring matches with those

found in aNySIM (better seen in figure 6.8), but the underlying model still does not capture the mooring tensions for the higher frequencies. Attempts have been made to include a lumped-mass approach, but this method is not fully incorporated in aNySIM fredom yet and gave unreliable results. Therefore, it is chosen to run the aNySIM TD simulations with a quasi-static mooring modelling, so a better comparison between the models can be made. Figure E.1 does include an OpenFAST simulation wherein the lumped-mass approach was applied, showing the large differences between a lumped-mass model and a quasi-static model just like was the case in figure 3.11. In section 3.2.2 however, it is mentioned that nevertheless the quasi-static approach is sufficient for the initial stage optimization.

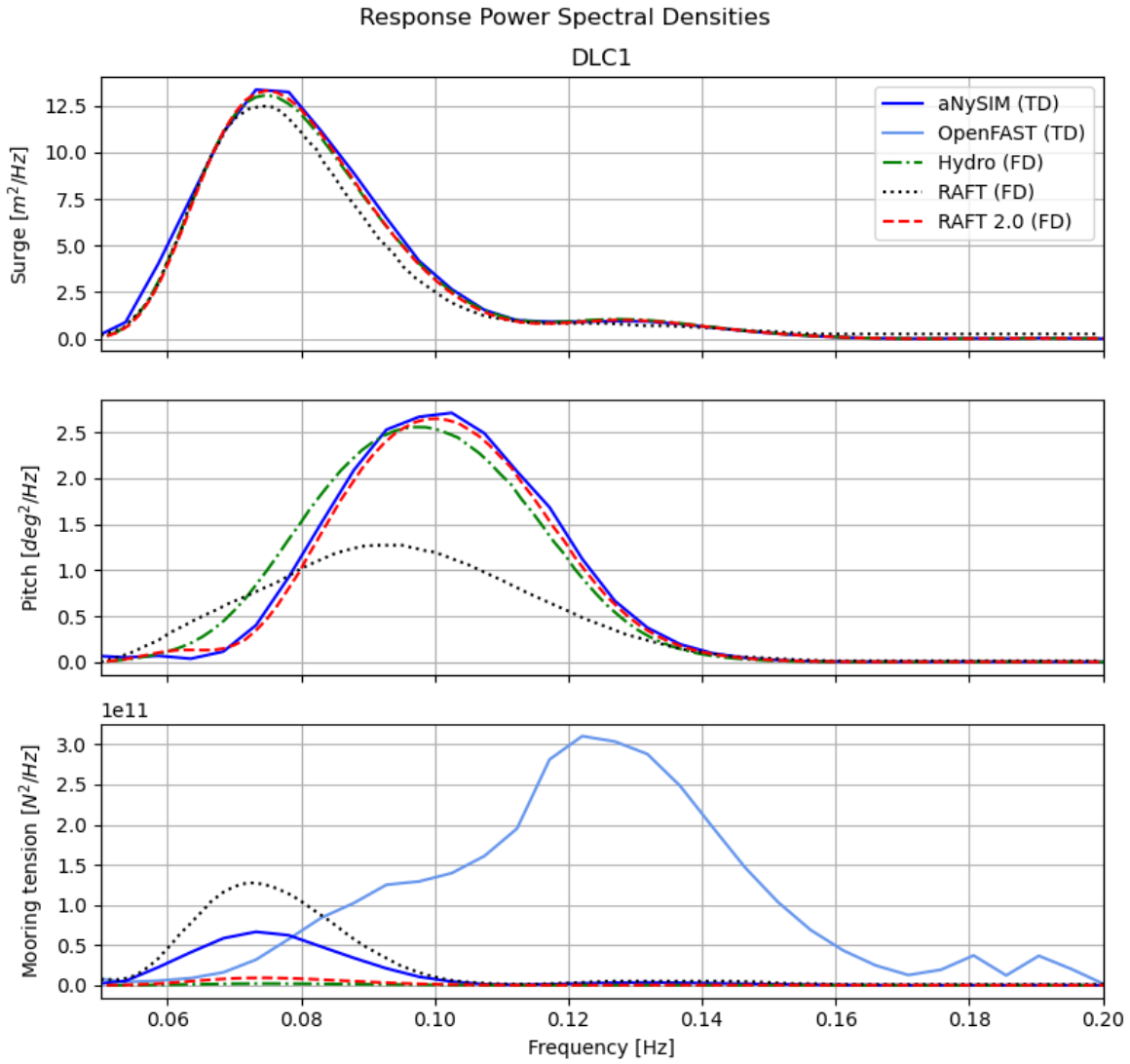

Figure 6.7: Power spectral density response for surge, pitch and mooring tension, for DLC 1, showing the time-domain model results of aNySIM, as well as the frequency-domain model results from RAFT, RAFT 2.0, and the hydro approach

Somewhat the same conclusions can be drawn from the results for the other load cases. The responses for the second, third, and fourth DLC are given in figure 6.8. The differences between the predictions made using RAFT 2.0 and the hydro approach in surge direction are small. However, both over predict the surge response at their peaks for the two load cases where the aerodynamics are relatively important, i.e. the above rated wind cases. Perhaps the one-on-one conversion from a rotational velocity feedback to a translational velocity feedback downplays the feedback action too much to implement it as done in the model. Being able to simulate the aerodynamic excitation could help give a better insight in examining the applied nacelle velocity feedback method, as the gain is the most active in the lower frequency range. However, the current method does ensure a nice fit for

the pitch response. Especially in the above rated wind cases it is shown to improve the pitch motion prediction. The motion is damped due to the contribution from the aerodynamic coefficients, which also can be seen in figure 6.4.

In terms of the mooring line tensions, the conclusions for these load cases align with the one for the first design load case: The mooring linearization process does ensure that the frequency peaks and frequency range for the mooring matches with those found in aNySIM, but under predict the tensions and it does not capture the mooring tensions for the higher frequencies. It is expected that a lumped-mass approach would improve on these predictions. However, the required computation time of the lumped mass method tends to be significant compared with the frequency-domain analysis [117].

The responses for each design load case are also depicted in appendix E, in a larger size.

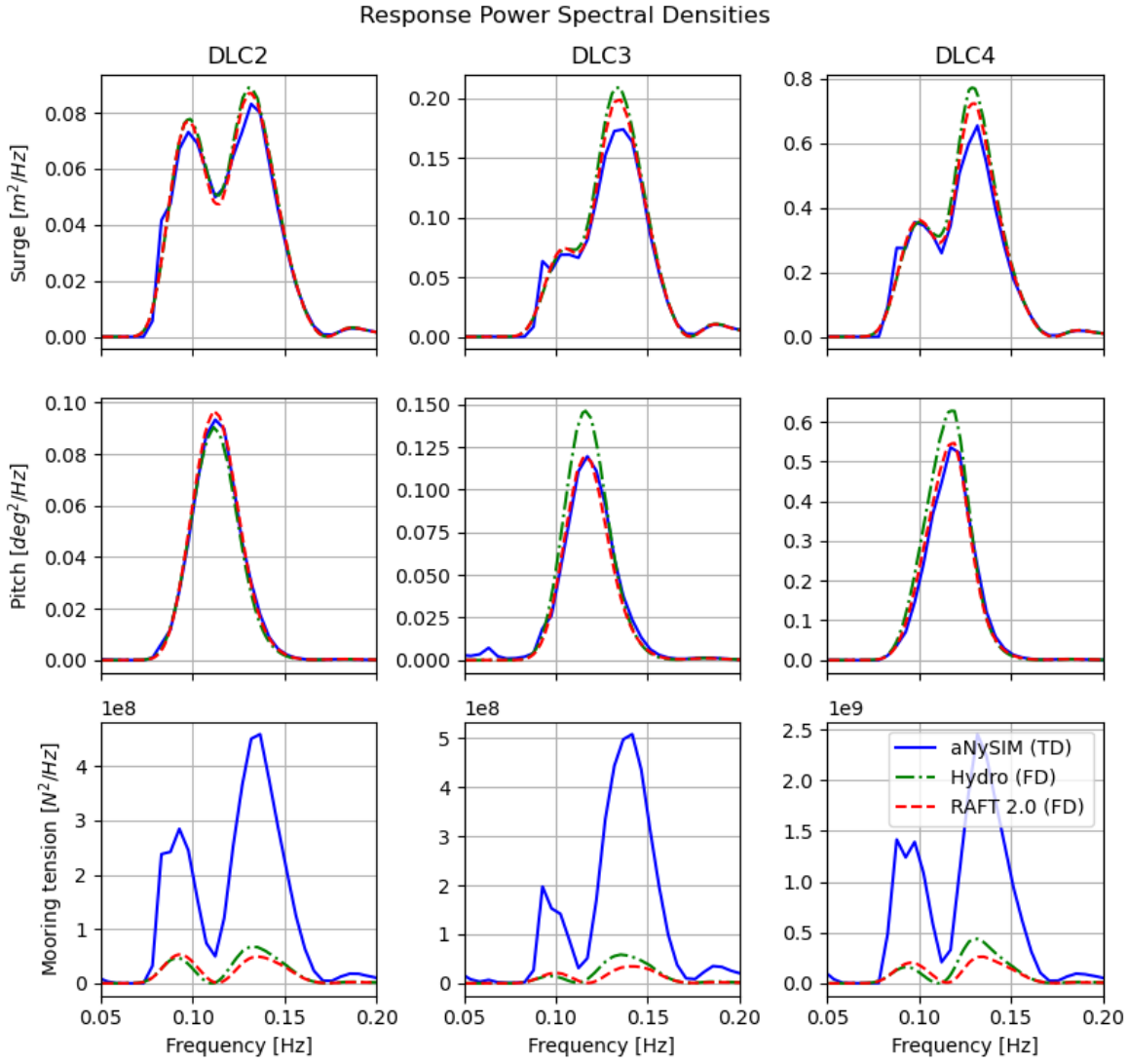

Figure 6.8: Power spectral density response for surge, pitch and mooring tension, for DLC 2, 3, and 4, showing the time-domain model results of aNySIM, as well as the frequency-domain model results from RAFT 2.0 and the hydro approach. Larger versions of each DLC are presented in appendix E

6.3. Recommendations

Two recommendations are a direct result of the simulation results shown in section 6.2.2: The mooring linearization process needs to be improved, and the low frequency range wind excitation must be added in the final solver.

The attempt of linearizing the mooring response using aNySIM fredom as done during this thesis did not turn out to be improving the model. Applying a lumped-mass model is expected to improve these results, although it might add computational time, since it is better capable of capturing the mooring lines' drag and inertia at higher frequencies.

Concerning the low frequency range wind excitation, the adapted Rotor class is already capable of determining the wind excitation, but the solver used to determine the final motions is not. The solver determines the motions using:

$$\vec{x}(\omega) = \frac{\vec{F}(\omega)}{-\omega^2 \left(\mathbf{M} + \mathbf{A}(\omega)\right) + i\omega\mathbf{B}(\omega) + \mathbf{C}(\omega)}$$

where $\vec{F}(\omega)$ now only consists of the first and second order wave loads, read from the hyd-file, in combination with the specified wave spectrum. This should be extended to include the aerodynamic excitation, so that $\vec{F}(\omega) = \vec{F_{hydro}}(\omega)$ + $\vec{F_{aero}}(\omega)$. This could be even further improved by including the floater motion induced wind velocities in the wind spectrum as shown in figure 6.1. Being able to examine the motions in the lower frequency range would improve on the verification of the applied aerodynamics.

Furthermore, as mentioned in section 5.3, the viscous damping of the floater is currently included by using given data. If this data is not know yet, an estimation of this viscous damping should be made to include it in the model. Including a method to do so would improve the FD model, since it would make it possible to test all kinds of floater designs in a cost and time efficient manner. To do so, one could e.g. apply the method proposed by Housseine, Monroy, and Hauteclocque [51].

If the three above mentioned recommendations are implemented in the model, the model can e.g. be used for sizing studies of existing FOWT designs. Both for the case when the design need to be sized in order to fit a larger turbine, as well as for a design that is optimized for certain site specific conditions and thereby needs sizing to keep the responses in different environmental conditions the same at hub height.

Lastly, since all recent projects have in common that they have a trend towards larger turbine sizes over the years [56] as shown in section 2.3.4, adding structural flexibility becomes more important, since the large turbines are becoming increasingly tall and more flexible [41]. The current model assumes that the whole design is a rigid body, thus also the turbine and blades. Flexibility has a noticeable impact on the floater motions and mooring line tensions [4]. Therefore, avoiding this assumption in the model would improve it.

7

Conclusion

The goal of this thesis was to develop and validate a cost efficient tool to evaluate FOWT designs, consisting out of both the turbine its aerodynamic contributions, including control, and the floater its hydrodynamic contributions.

To achieve this, a Python wrapper was built to use DIFFRAC and aNySIM fredom for the calculations of the hydrodynamics and RAFT's Rotor class for the aerodynamics. The latter needed some adjustments to include the nacelle velocity feedback gain. It is shown that applying BEM calculations for the hydrodynamics already improves the FOWT's motion predictions compared to the 2D strip-theory applied in RAFT, but that including the aerodynamic coefficients improves the model even further, especially for the pitch motion.

The mooring line tension predictions however, did not improve. Linearizing the response in aNySIM based on the velocity standard deviations turned out to be insufficient.

Therefore, to develop the model it is suggested to adopt a lumped-mass mooring line model to improve the mooring tension predictions. Furthermore, the low frequency range wind excitation must be added in the final solver, in order for the model to be able to predict the motions due to the wind excitation in this frequency range.

References

[1] 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.

[2] Adnan Memija.

[3] Christopher Allen et al. *Definition of the UMaine VolturnUS-S Reference Platform Developed for the IEA Wind 15-Megawatt Offshore Reference Wind Turbine*. Golden, United States: National Renewable Energy Laboratory, 2020.

[4] Morten Thøtt Andersen. “Floating foundations for offshore wind turbines”. In: (2016).

[5] Erin E Bachynski et al. “Real-time hybrid model testing of a braceless semi-submersible wind turbine: Part II—Experimental results”. In: *International Conference on Offshore Mechanics and Arctic Engineering*. Vol. 49972. American Society of Mechanical Engineers. 2016, pp. 1–12.

[6] Erin Elizabeth Bachynski. “Design and dynamic analysis of tension leg platform wind turbines”. PhD thesis. NTNU, 2014. ISBN: 978-82-326-0096-0.

[7] Christian Bak et al. “The DTU 10-MW reference wind turbine”. In: *Danish wind power research 2013*. 2013.

[8] Blue H Engineeringl. *Historical development*. 2023. URL: `http://www.bluehengineering.com/historical-development.html` (visited on 04/05/2023).

[9] Bluewater. *Flaoting Offshore Wind Tension Leg Platform*. 2021. URL: `https://www.bluewater.com/our-solutions/renewable-energy-solutions/` (visited on 04/04/2023).

[10] Michael Borg and Maurizio Collu. “A comparison between the dynamics of horizontal and vertical axis offshore floating wind turbines”. In: *Phil.Trans.R.Soc.A.* 373.20140076 (2015), pp. 1–16.

[11] EA Bossanyi. “Wind turbine control for load reduction”. In: *Wind Energy: An International Journal for Progress and Applications in Wind Power Conversion Technology* 6.3 (2003), pp. 229–244.

[12] Djillali Amar Bouzid, Subhamoy Bhattacharya, and Lalahoum Otsmane. “Assessment of natural frequency of installed offshore wind turbines using nonlinear finite element model considering soil-monopile interaction”. In: *Journal of Rock Mechanics and Geotechnical Engineering* 10.2 (2018), pp. 333–346.

[13] Marshall L Buhl Jr. *New empirical relationship between thrust coefficient and induction factor for the turbulent windmill state*. Tech. rep. Golden, United States, 2005.

[14] Sandy Butterfield et al. *Engineering challenges for floating offshore wind turbines*. Tech. rep. Golden, United States, 2007.

[15] BVG Associates Limited.

[16] BW Ideol. *A Flaoting Barge*. 2023. URL: `https://www.bw-ideol.com/en/technology` (visited on 04/04/2023).

[17] Lichao Cao et al. “Wind farm layout optimization to minimize the wake induced turbulence effect on wind turbines”. In: *Applied Energy* 323 (2022), p. 119599.

[18] Lucas Henrique Souza do Carmo. “A slender-body approach for computing second-order wave forces in seakeeping simulations of floating offshore wind turbines”. PhD thesis. Universidade de São Paulo, 2021.

[19] Laura Castro-Santos and Vicente Diaz-Casas. *Floating offshore wind farms*. Springer, 2016.

[20] Subrata Kumar Chakrabarti. *Offshore structure modeling*. Vol. 9. world scientific, 1994.

[21] Gustave Paul Corten. “Flow separation on wind turbine blades”. In: *University of Utrecht* 733 (2001).

[22] Abdeldjalil Dahbi, Nasreddine Nait-Said, and Mohamed-Said Nait-Said. “A novel combined MPPT-pitch angle control for wide range variable speed wind turbine based on neural network”. In: *International Journal of Hydrogen Energy* 41.22 (2016), pp. 9427–9442.

[23] Carlos Eduardo Silva de Souza. “Structural modelling, coupled dynamics, and design of large floating wind turbines”. PhD thesis. NTNU, 2022. ISBN: 978-82-326-5375-1.

[24] Carlos Eduardo Silva de Souza and Erin E Bachynski. “Changes in surge and pitch decay periods for floating wind turbines for varying wind speed”. In: *Ocean Engineering* 180 (2019), pp. 223–237.

[25] Carlos Eduardo Silva de Souza, John Marius Hegseth, and Erin E Bachynski. “Frequency–Dependent Aerodynamic Damping and Inertia in Linearized Dynamic Analysis of Floating Wind Turbines”. In: *Journal of Physics: Conference Series* 1452.012040 (2020), pp. 1–13.

[26] Det Norske Veritas. *Environmental Conditions And Environdmental Loads*. Norway: DNV, 2021.

[27] Frederick Driscoll et al. “Validation of a FAST model of the statoil-hywind demo floating wind turbine”. In: *Energy Procedia* 94 (2016), pp. 3–19.

[28] Katherine Dykes et al. *Results of IEA Wind TCP Workshop on a Grand Vision for Wind Energy Technology*. IEA Wind TCP, 2019.

[29] First Edition, JMJ Journée, and WW Massie. “Offshore hydromechanics”. In: *Delft University of Technology* (2001).

[30] Equinor. *Hywind-Scotland*. 2017. URL: `https://www.equinor.com/energy/hywind-scotland` (visited on 04/04/2023).

[31] Equinor. *Hywind-Tampen*. 2022. URL: `https://www.equinor.com/energy/hywind-tampen` (visited on 04/04/2023).

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

[33] Boris Fischer and Martin Shan. “A survey on control methods for the mitigation of tower loads”. In: *Fraunhofer IWES Report* 655 (2013).

[34] FLAGSHIP. *The FLAGSHIP concept*. 2020. URL: `https://www.flagshiproject.eu/the-project/` (visited on 04/04/2023).

[35] Evan Gaertner et al. *Definition of the IEA 15-Megawatt Offshore Reference Wind Turbine*. Golden, United States: National Renewable Energy Laboratory, 2020.

[36] Global Wind Energy Council. *Floating Offshore Wind - A Global Opportunity*. Global Wind Energy Council, 2022.

[37] Global Wind Energy Council. *Global Wind Report 2022*. Brussels, Belgium: Global Wind Energy Council, 2022.

[38] Andrew J Goupee, Richard W Kimball, and Habib J Dagher. “Experimental observations of active blade pitch and generator control influence on floating wind turbine response”. In: *Renewable Energy* 104 (2017), pp. 9–19.

[39] Andrew J. Goupee et al. “Experimental comparison of three floating wind turbine concepts”. In: *Journal of Offshore Mechanics and Arctic Engineering* 136.2 (2014).

[40] Ozan Gözcü, Stavros Kontos, and Henrik Bredmose. “Dynamics of two floating wind turbines with shared anchor and mooring lines”. In: *Journal of Physics: Conference Series*. Vol. 2265. 4. IOP Publishing. 2022, p. 042026.

[41] Chaojie Gu et al. “Dynamic analysis of flexible wind turbine tower by a transfer matrix method”. In: *International Journal of Structural Stability and Dynamics* 21.10 (2021), p. 2150142.

[42] S Gueydon, I Bayati, and EJ De Ridder. “Discussion of solutions for basin model tests of FOWTs in combined waves and wind”. In: *Ocean Engineering* 209 (2020), p. 107288.

[43] GustoMSC. *Tri-Floater concept*. 2023. URL: `https://www.nov.com/products/tri-floater-floating-offshore-wind-turbine-foundation` (visited on 04/04/2023).

[44] GVPM Politecnico Milano. *Floating Offshore Wind Turbines*. 2015. URL: https://www.windtunnel.polimi.it/aeroelasticity/floating-offshore-wind-turbine/ (visited on 11/13/2023).

[45] Matthew Hall and Andrew Goupee. "Validation of a lumped-mass mooring line model with DeepCwind semisubmersible model test data". In: *Ocean Engineering* 104 (2015), pp. 590–603.

[46] Matthew Hall et al. "An Open-Source Frequency-Domain Model for Floating Wind Turbine Design Optimization". In: *Journal of Physics: Conference Series*. Vol. 2265. 042020. IOP Publishing, 2022, pp. 1–12.

[47] Mary Harvey et al. *Floating Wind Joint Industry Programme: Phase IV Summary Report*. London, United Kingdom: The Carbon Trust, 2022.

[48] Klaus Hasselmann et al. "Measurements of wind-wave growth and swell decay during the Joint North Sea Wave Project (JONSWAP)." In: *Ergaenzungsheft zur Deutschen Hydrographischen Zeitschrift, Reihe A* (1973).

[49] John Marius Hegseth and Erin E Bachynski. "A semi-analytical frequency domain model for efficient design evaluation of spar floating wind turbines". In: *Marine Structures* 64 (2019), pp. 186–210.

[50] Hexicon. *TwinWind: Cutting Edge Technology*. 2022. URL: https://www.hexicongroup.com/twinwind/ (visited on 04/04/2023).

[51] Charaf Ouled Housseine, Charles Monroy, and Guillaume de Hauteclocque. "Stochastic linearization of the Morison equation applied to an offshore wind turbine". In: *Proceedings of the International Conference on Offshore Mechanics and Arctic Engineering, St. John's, NL, Canada*. Vol. 9. 2015.

[52] IEA. *UMaine semi-submersible DISCON.IN*. 2023. URL: https://github.com/IEAWindTask37/IEA-15-240-RWT/blob/master/OpenFAST/IEA-15-240-RWT-UMaineSemi/IEA-15-240-RWT-UMaineSemi_DISCON.IN (visited on 11/18/2023).

[53] International Energy Agency. *IEA Wind Task 37*. 2023. URL: https://github.com/IEAWindTask37/IEA-22-280-RWT (visited on 04/05/2023).

[54] International Energy Agency. *IEA Wind Task 37*. 2023. URL: https://github.com/IEAWindTask37/IEA-15-240-RWT (visited on 06/05/2023).

[55] International Energy Agency. *Net Zero by 2050. A Roadmap for the Global Energy Sector*. Paris, France: IEA Publications, 2021.

[56] IRENA. *Global Renewables Outlook: Energy transformation 2050 (Edition: 2020)*. International Renewable Energy Agency, Abu Dhabi. ISBN: 978-92-9260-238-3.

[57] Rhodri James and Marc Costa Ros. *Floating Offshore Wind: Market and Technology Review*. London, United Kingdom: The Carbon Trust, 2015.

[58] Nick Jenkins et al. *Wind Energy Handbook*. Third Edition. John Wiley & Sons, 2021.

[59] Jason Jonkman. *Definition of the Floating System for Phase IV of OC3*. Golden, United States: National Renewable Energy Laboratory, 2010.

[60] Jason Jonkman. *Dynamics Modeling and Loads Analysis of an Offshore Floating Wind Turbine*. Golden, United States: National Renewable Energy Laboratory, 2007.

[61] Jason Jonkman. "Influence of control on the pitch damping of a floating wind turbine". In: *46th AIAA aerospace sciences meeting and exhibit*. 2008, p. 1306.

[62] Jason Jonkman. "The new modularization framework for the FAST wind turbine CAE tool". In: *51st AIAA aerospace sciences meeting including the new horizons forum and aerospace exposition*. 2013, p. 202.

[63] Jason Jonkman et al. "Definition of a 5-MW Reference Wind Turbine for Offshore System Development". In: *Tech.Rep. NREL/TP-500-38060*. Golden, United States: National Renewable Energy Laboratory, 2009.

[64] Min-Jae Kang and Ho-Chan Kim. “Neural network based pitch controller”. In: *Proceedings of the World Congress on Electrical Engineering and Computer Systems and Science, Barcelona, Spain*. 2015, pp. 140–146.

[65] Meysam Karimi, Brad Buckham, and Curran Crawford. “A fully coupled frequency domain model for floating offshore wind turbines”. In: *Journal of Ocean Engineering and Marine Energy* 5 (2019), pp. 135–158.

[66] Matthew A. Lackner. “An investigation of variable power collective pitch control for load mitigation of floating offshore wind turbines”. In: *Wind energy* 16.3 (2013), pp. 435–444.

[67] Torben J. Larsen and T.D. Hanson. “A method to avoid negative damped low frequent tower vibrations for a floating, pitch controlled wind turbine”. In: *Journal of Physics: Conference Series*. Vol. 75. 012073. IOP Publishing, 2007, pp. 1–11.

[68] Mareike Leimeister, A. Kolios, and M. Collu. “Critical review of floating support structures for offshore wind farm deployment”. In: *Journal of Physics: Conference Series*. Vol. 1104. 012007. IOP Publishing, 2018, pp. 1–11.

[69] Ericka Lozon and Matthew Hall. “Coupled loads analysis of a novel shared-mooring floating wind farm”. In: *Applied Energy* 332 (2023), p. 120513.

[70] RC Lupton. “Frequency-domain modelling of floating wind turbines”. PhD thesis. University of Cambridge, 2015.

[71] Richard C. Lupton and Robin S. Langley. “Improved linearised models of wind turbine aerodynamics and control system dynamics using harmonic linearisation”. In: *Renewable Energy*. Vol. 135. 2019, pp. 148–162.

[72] Kai-Tung Ma et al. *Mooring system engineering for offshore structures*. Gulf Professional Publishing, 2019.

[73] Michel Make. “Predicting scale effects on floating offshore wind turbines”. In: *MSc Thesis, Confidential* (2014).

[74] James F. Manwell, Jon G. McGowan, and Anthony L. Rogers. *Wind Energy Explained: Theory, Design and Application*. Second edition. John Wiley & Sons, Ltd., 2009. ISBN: 978-11-1999-436-7.

[75] MARIN. *aNySIM XMF*. 2023. URL: `https://www.marin.nl/en/about/facilities-and-tools/software/anysim` (visited on 11/17/2023).

[76] MARIN. *DIFFRAC*. 2023. URL: `https://www.marin.nl/en/about/facilities-and-tools/software/diffrac` (visited on 11/17/2023).

[77] MARIN. *Software*. 2023. URL: `https://www.marin.nl/en/about/facilities-and-tools/software` (visited on 11/17/2023).

[78] David Marten and Juliane Wendler. “Qblade guidelines”. In: *Ver. 0.6, Technical University of (TU Berlin), Berlin, Germany* (2013).

[79] Denis Matha, Frank Sandner, and David Schlipf. “Efficient critical design load case identification for floating offshore wind turbines with a reduced nonlinear model”. In: *Journal of Physics: Conference Series*. Vol. 555. 1. IOP Publishing. 2014, p. 012069.

[80] J McMorland et al. “Operation and maintenance for floating wind turbines: A review”. In: *Renewable and Sustainable Energy Reviews* 163 (2022), p. 112499.

[81] Patrick J Moriarty and A Craig Hansen. *AeroDyn theory manual*. Tech. rep. National Renewable Energy Lab., Golden, CO (US), 2005.

[82] Sebastiaan Paul Mulders et al. “Preventing wind turbine tower natural frequency excitation with a quasi-LPV model predictive control scheme”. In: *Wind Energy* 23.3 (2020), pp. 627–644.

[83] Walter Musial et al. *Offshore Wind Market Report: 2022 Edition*. Tech. rep. Golden, United States, 2022.

[84] John Nicholas Newman. “Second-order slowly varying forces on vessels in irregular waves”. In: *Proceedings of the international symposium on dynamics of marine vehicles and structures in waves. London, UK, 1974*. 1974.

[85] S Andrew Ning. “A simple solution method for the blade element momentum equations with guaranteed convergence”. In: *Wind Energy* 17.9 (2014), pp. 1327–1345.

[86] NREL. *MoorPy (Quasi-Static Mooring Analysis in Python)*. 2021. URL: https://moorpy.readthedocs.io/en/latest/ (visited on 03/28/2023).

[87] NREL. *OpenFAST*. 2023. URL: https://openfast.readthedocs.io/en/main/index.html (visited on 04/05/2023).

[88] NREL. *RAFT version 1.0*. 2021. URL: https://openraft.readthedocs.io/en/latest/index.html (visited on 02/28/2023).

[89] NREL. *ROSCO version 2.6.0*. 2022. URL: https://rosco.readthedocs.io/en/latest/index.html (visited on 03/14/2023).

[90] NREL WISDEM. *CCBlade*. 2019. URL: https://wisdem.readthedocs.io/en/master/wisdem/ccblade/index.html (visited on 03/16/2023).

[91] OpenMDAO Development Team. *Wind Turbine Actuator Disc*. 2022. URL: https://openmdao.org/newdocs/versions/latest/examples/betz_limit.html (visited on 11/15/2023).

[92] Aldert Otter et al. “A review of modelling techniques for floating offshore wind turbines”. In: *Wind Energy* 25.5 (2022), pp. 831–857.

[93] 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.

[94] LAURA ROLO Perez. “Design, testing and validation of a scale model semisubmersible offshore wind turbine under regular/irregular waves and wind loads”. PhD thesis. University of Strathclyde Glasgow, UK, 2014.

[95] Principle Power. *OpenFAST*. 2022. URL: https://www.principlepower.com/projects/kincardine-offshore-wind-farm (visited on 04/06/2023).

[96] Principle Power. *WindFloat concept*. 2022. URL: https://www.principlepower.com/windfloat (visited on 04/04/2023).

[97] Dongsheng Qiao, Jinping Ou, and Fei Wu. “Design selection analysis for mooring positioning system of deepwater semi-submersible platform”. In: *ISOPE International Ocean and Polar Engineering Conference*. ISOPE. 2012, ISOPE–I.

[98] Renews.biz. *Kincardine facing turbine repair job*. 2022. URL: https://renews.biz/77609/kincardine-facing-turbine-repair-job/ (visited on 02/27/2023).

[99] Irene Rivera Arreba. “Computation of nonlinear wave loads on floating structures”. MA thesis. NTNU, 2017.

[100] Amy Robertson et al. *Definition of the semisubmersible floating system for phase II of OC4*. Golden, United States: National Renewable Energy Laboratory, 2014.

[101] Keyvan Sadeghi and Atilla Incecik. “Tensor properties of added-mass and damping coefficients”. In: *Journal of engineering mathematics* 52 (2005), pp. 379–387.

[102] Tom Salic et al. “Control Strategies for Floating Offshore Wind Turbine: Challenges and Trends”. In: *Electronics* 8.1185 (2019), pp. 1–14.

[103] SBM Offshore. *Float4WindTM TLP concept*. 2022. URL: https://www.sbmoffshore.com/newsroom/news-events/sbm-offshore-launches-float4windtm (visited on 04/04/2023).

[104] SeaTwirl. *The SeaTwirl concept*. 2023. URL: https://seatwirl.com/products/ (visited on 04/04/2023).

[105] Ahmad Sedaghat et al. “Determination of rated wind speed for maximum annual energy production of variable speed wind turbines”. In: *Applied energy* 205 (2017), pp. 781–789.

[106] Nataliia Y. Sergiienko et al. “Review of scaling laws applied to floating offshore wind turbines”. In: *Renewable and Sustainable Energy Reviews* 162.112477 (2022), pp. 1–13.

[107] Alexandre N Simos et al. “Slow-drift of a floating wind turbine: An assessment of frequency-domain methods based on model tests”. In: *Renewable energy* 116 (2018), pp. 133–154.

[108] Senu Sirnivas et al. “Coupled mooring analyses for the wec-sim wave energy converter design tool”. In: *International Conference on Offshore Mechanics and Arctic Engineering*. Vol. 49972. American Society of Mechanical Engineers. 2016, V006T09A023.

[109] Bjørn Skaare, Tor David Hanson, and Finn Gunnar Nielsen. “Importance of control strategies on fatigue life of floating wind turbines”. In: *International Conference on Offshore Mechanics and Arctic Engineering*. Vol. 42711. 2007, pp. 493–500.

[110] Bjørn Skaare et al. “Integrated Dynamic Analysis of Floating Offshore Wind Turbines”. In: *European Wind Energy Conference and Exhibition* (2007), pp. 1–11.

[111] Andrew Z.P. Smith. *UK offshore wind capacity factors*. 2023. URL: `https://energynumbers.info/uk-offshore-wind-capacity-factors` (visited on 02/22/2023).

[112] Carlos Eduardo S Souza, John Marius Hegseth, and Erin E Bachynski. “Frequency-dependent aerodynamic damping and inertia in linearized dynamic analysis of floating wind turbines”. In: *Journal of Physics: Conference Series*. Vol. 1452. 1. IOP Publishing. 2020, p. 012040.

[113] Carlos Eduardo Silva de Souza and Erin E Bachynski-Polić. “Design, structural modeling, control, and performance of 20 MW spar floating wind turbines”. In: *Marine Structures* 84 (2022), p. 103182.

[114] Dan Kyle Spearman and Sam Strivens. *Floating Wind Joint Industry Project: Phase II Summary Report*. London, United Kingdom: The Carbon Trust, 2020.

[115] Joep van der Spek. “The response of a semi-submersible floating offshore wind turbine under misaligned wind and waves”. MA thesis. Delft University of Technology, 2022.

[116] Stiesdal. *Tetra offshore foundations for any water depth*. 2023. URL: `https://www.stiesdal.com/offshore/tetra-offshore-foundations-for-any-water-depth/` (visited on 04/04/2023).

[117] Imanol Touzon et al. “Frequency domain modelling of a coupled system of floating structure and mooring Lines: An application to a wave energy converter”. In: *Ocean Engineering* 220 (2021), p. 108498.

[118] Axelle Viré and Delphine De Tavernier. *AE4W31 Floating Offshore Wind Energy: Lecture slides*. 2022.

[119] Julian Wolfram. “On alternative approaches to linearization and Morison’s equation for wave forces”. In: *Proceedings of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences* 455.1988 (1999), pp. 2957–2974.

[120] Dan Zhao et al. “Chapter 5: Wind turbines and aerodynamics energy harvesters”. In: *Offshore wind turbine aerodynamics modelling and measurements*. Academic Press, 2019, pp. 373–400.

A

General arrangement

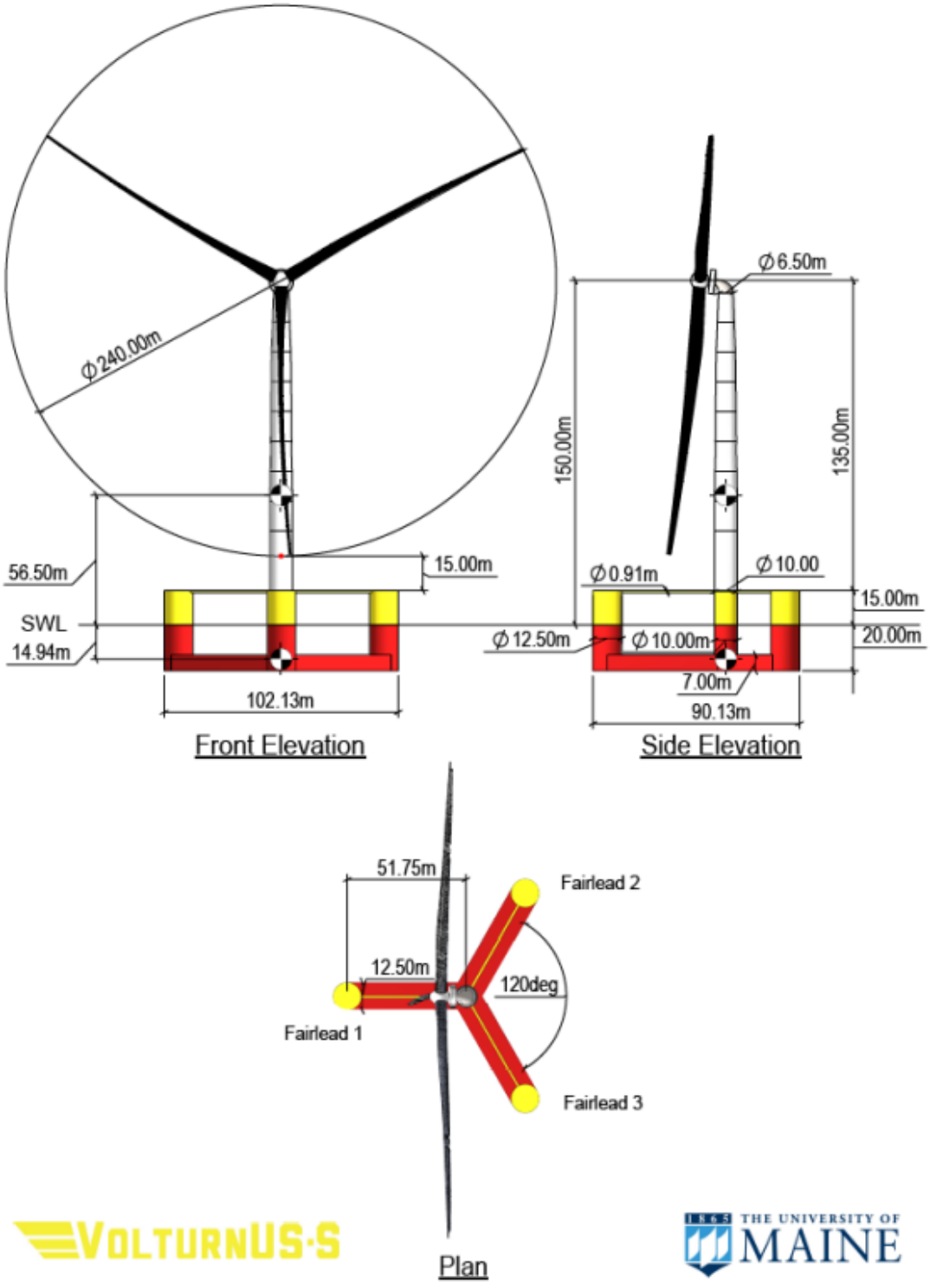

Figure A.1: The general arrangement of UMain's VolturnUS-S semi-submersible design (retrieved from [3])

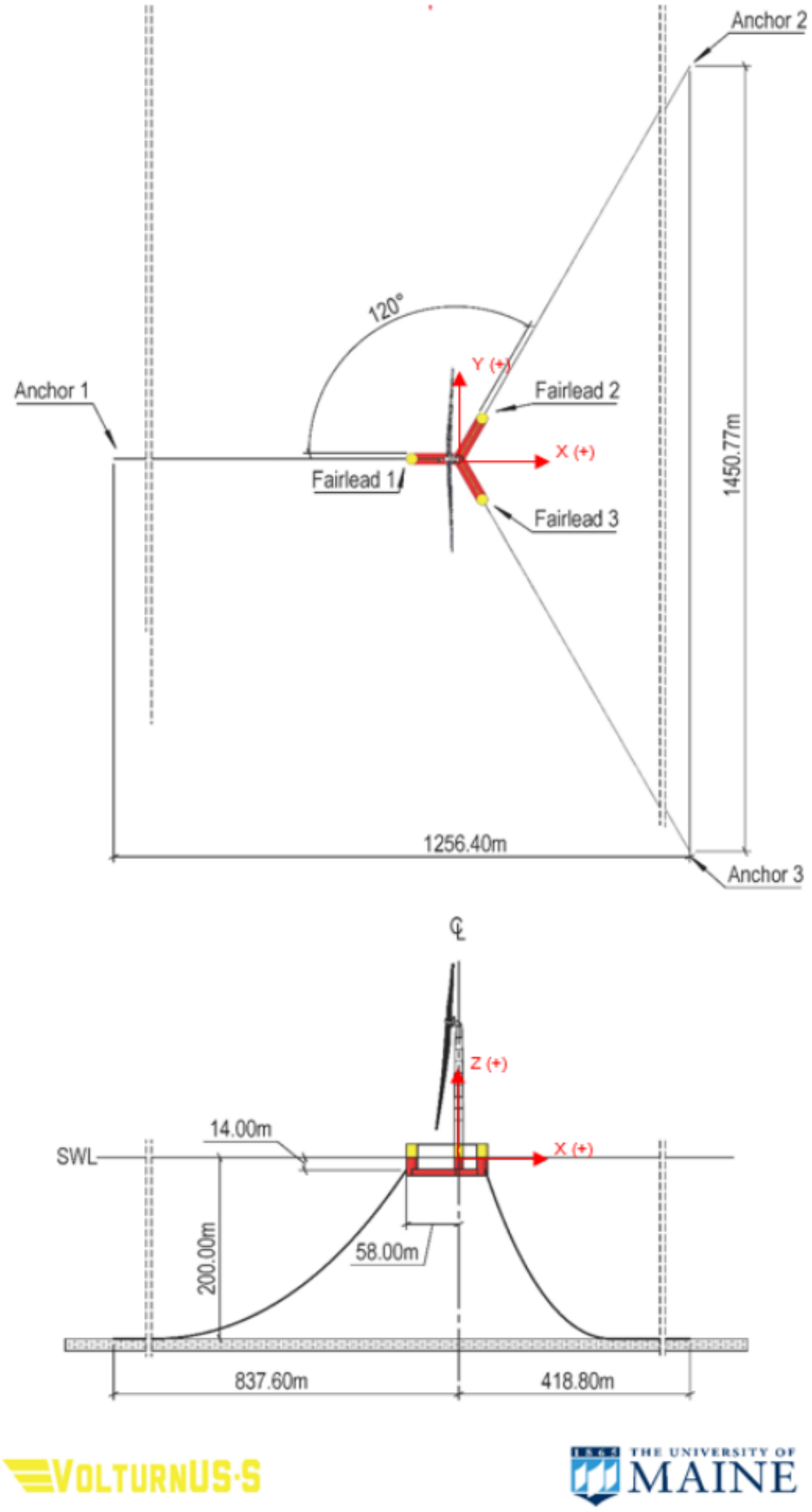

Figure A.2: Mooring system arrangement of UMain's VolturnUS-S semi-submersible design (retrieved from [3])

B

Turbine YAML

Listed below is the yaml-file used to describe the rotor design and control settings. The ”FI_Kp” values at the end are added to the file, and the first three entries are copied from the ”site” description. RAFT manually adds them to the yaml-file in Python, by copying them from the site description, I chose to add them in the yaml-file directly. As the lists for e.g. airfoil data are quite long, I have shortened the lists with ”...” in the appendix to keep it more compact. The complete file has the last line coded at line 1062, rather than 146. The missing values can be found on RAFT’s GitHub page.

```
type: input file for RAFT
name: IEA 15 MW with VolturnUS-S steel semi

turbine:
    rho_air     : 1.225      # [kg/m^3] air density                    ##### COPIED FROM "SITE" TO MAKE RAFT_ROTOR WORK, NEEDED TO BE INSIDE "TURBINE"
    mu_air      : 1.81e-05   #          air dynamic viscosity          ##### COPIED FROM "SITE" TO MAKE RAFT_ROTOR WORK, NEEDED TO BE INSIDE "TURBINE"
    shearExp    : 0          #    used to be 0.12      shear exponent   ##### COPIED FROM "SITE" TO MAKE RAFT_ROTOR WORK, NEEDED TO BE INSIDE "TURBINE"

    mRNA          :     991000        #  [kg]       RNA mass
    IxRNA         :          0        #  [kg-m2]    RNA moment of inertia about local x axis (assumed to be identical to rotor axis for now, as approx) [kg-m^2]
    IrRNA         :          0        #  [kg-m2]    RNA moment of inertia about local y or z axes [kg-m^2]
    xCG_RNA       :          0        #  [m]        x location of RNA center of mass [m] (Actual is ~= -0.27 m)
    hHub          :        150.0      #  [m]        hub height above water line [m]
    Fthrust       :       1500.0E3    #  [N]        temporary thrust force to use

    I_drivetrain: 318628138.0   # full rotor + drivetrain inertia as felt on the high-speed shaft

    nBlades     : 3       # number of blades
    Zhub        : 150.0   # hub height [m]
    Rhub        : 3.97    # hub radius [m]
    precone     : 4.0     # [deg]
    shaft_tilt  : 6.0     # [deg]
    overhang    : 12.0313 # [m]
    aeroServoMod : 2      # 0 aerodynamics off; 1 aerodynamics on (no control); 2 aerodynamics and control on

    blade:
        precurveTip : -3.9999999999999964   #
        presweepTip : 0.0                   #
        Rtip        : 120.96999999936446    # rotor radius

        #           r         chord      theta      precurve     presweep
        geometry:
          - [     8.004,      5.228,     15.474,      0.035,      0.000 ]
          - [    12.039,      5.321,     14.692,      0.084,      0.000 ]
```

```
        - [    16.073,      5.458,     13.330,      0.139,      0.000 ]
        ...
        - [   108.867,      2.283,     -2.108,     -2.864,      0.000 ]
        - [   112.901,      2.096,     -1.953,     -3.224,      0.000 ]
        - [   116.936,      1.902,     -1.662,     -3.605,      0.000 ]
      #    station(rel)      airfoil name
      airfoils:
        - [   0.00000, circular ]
        - [   0.02000, circular ]
        - [   0.15000, SNL-FFA-W3-500 ]
        - [   0.24517, FFA-W3-360 ]
        - [   0.32884, FFA-W3-330blend ]
        - [   0.43918, FFA-W3-301 ]
        - [   0.53767, FFA-W3-270blend ]
        - [   0.63821, FFA-W3-241 ]
        - [   0.77174, FFA-W3-211 ]
        - [   1.00000, FFA-W3-211 ]

  airfoils:
    - name               : circular  #
      relative_thickness : 1.0  #
      data:  # alpha        c_l         c_d         c_m
        - [ -179.9087,    0.00010,    0.35000,   -0.00010 ]
        - [  179.9087,    0.00010,    0.35000,   -0.00010 ]
    - name               : SNL-FFA-W3-500  #
      relative_thickness : 0.5  #
      data:  # alpha        c_l         c_d         c_m
        - [ -179.9660,    0.00000,    0.08440,    0.00000 ]
        - [ -170.0000,    0.44190,    0.08440,    0.31250 ]
        - [ -160.0002,    0.88370,    0.12680,    0.28310 ]
        ...
        - [  160.0002,   -0.88370,    0.12680,   -0.53250 ]
        - [  170.0000,   -0.44180,    0.08440,   -0.39060 ]
        - [  179.9660,    0.00000,    0.08440,    0.00000 ]
    - name               : FFA-W3-211  #
      relative_thickness : 0.211  #
      data:  # alpha        c_l         c_d         c_m
        - [ -179.9087,    0.00000,    0.02464,    0.00000 ]
        - [ -177.7143,    0.05403,    0.02534,    0.09143 ]
        - [ -175.4286,    0.10805,    0.02742,    0.18286 ]
        ...
        - [  175.4286,   -0.10805,    0.02742,   -0.22857 ]
        - [  177.7143,   -0.05403,    0.02534,   -0.11429 ]
        - [  179.9087,    0.00000,    0.02464,    0.00000 ]
    - name               : FFA-W3-241  #
      relative_thickness : 0.241  #
      data:  # alpha        c_l         c_d         c_m
        - [ -179.9087,    0.00000,    0.01178,    0.00000 ]
        - [ -177.7143,    0.05818,    0.01248,    0.09143 ]
        - [ -175.4286,    0.11636,    0.01460,    0.18286 ]
        ...
        - [  175.4286,   -0.11636,    0.01460,   -0.22857 ]
        - [  177.7143,   -0.05818,    0.01248,   -0.11429 ]
        - [  179.9087,    0.00000,    0.01178,    0.00000 ]
    - name               : FFA-W3-270blend  #
      relative_thickness : 0.27  #
      data:  # alpha        c_l         c_d         c_m
        - [ -179.9087,    0.00000,    0.01545,    0.00000 ]
        - [ -177.7143,    0.06213,    0.01611,    0.09143 ]
        - [ -175.4286,    0.12426,    0.01807,    0.18286 ]
        ...
        - [  175.4286,   -0.12426,    0.01807,   -0.22857 ]
        - [  177.7143,   -0.06213,    0.01611,   -0.11429 ]
        - [  179.9087,    0.00000,    0.01545,    0.00000 ]
    - name               : FFA-W3-301  #
      relative_thickness : 0.301  #
      data:  # alpha        c_l         c_d         c_m
        - [ -179.9087,    0.00000,    0.02454,    0.00000 ]
        - [ -177.7143,    0.06508,    0.02514,    0.09143 ]
        - [ -175.4286,    0.13016,    0.02694,    0.18286 ]
        ...
```

```
        - [  175.4286,   -0.13016,    0.02694,   -0.22857 ]
        - [  177.7143,   -0.06508,    0.02514,   -0.11429 ]
        - [  179.9087,    0.00000,    0.02454,    0.00000 ]
    - name               : FFA-W3-330blend  #
      relative_thickness : 0.33  #
      data:  # alpha        c_l         c_d         c_m
        - [ -179.9087,    0.00000,    0.03169,    0.00000 ]
        - [ -177.7143,    0.06960,    0.03228,    0.09143 ]
        - [ -175.4286,    0.13920,    0.03406,    0.18286 ]
        ...
        - [  175.4286,   -0.13920,    0.03406,   -0.22857 ]
        - [  177.7143,   -0.06960,    0.03228,   -0.11429 ]
        - [  179.9087,    0.00000,    0.03169,    0.00000 ]
    - name               : FFA-W3-360  #
      relative_thickness : 0.36  #
      data:  # alpha        c_l         c_d         c_m
        - [ -179.9087,    0.00000,    0.03715,    0.00000 ]
        - [ -177.7143,    0.07178,    0.03774,    0.09143 ]
        - [ -175.4286,    0.14356,    0.03951,    0.18286 ]
        ...
        - [  175.4286,   -0.14356,    0.03951,   -0.22857 ]
        - [  177.7143,   -0.07178,    0.03774,   -0.11429 ]
        - [  179.9087,    0.00000,    0.03715,    0.00000 ]

  pitch_control:
    GS_Angles: [0.062094,  0.088628,  0.109270,  ...,  0.377928,  0.385970,  0.393863]#[rad
    GS_Kp: [-1.143968,  -0.988219,  -0.859873,  ...,  0.054835,  0.065966,  0.076516] # [s]
    GS_Ki: [-0.119726,  -0.108178,  -0.098661,  ..., -0.030839,  -0.030013,  -0.029231]#[-]
    Fl_Kp: -0.195825942 # Used to be -9.35, but they set it to zero manually in their
        because "there seems to be a discrepancy with its definition"
    Fl_Kp_cutoff: 0.213 # [rad/s]
    Fl_Kp_damp_ratio: 1 # [-]
    Fl_Kp_notch: 3.12 # [rad/s]
    Fl_Kp_Qfactor: 2 # [-]
  wt_ops: # operating points: wind speed [m/s], blade pitch [deg], rotor speed [rpm]
      v: [3.0, 3.266896551724138, 3.533793103448276, ..., 24.016551724137933,
          24.508275862068963, 25.0]
      pitch_op: [-0.25, -0.25, -0.25, ..., 22.0067, 22.4925, 22.9724]
      omega_op: [2.1486, 2.3397, 2.5309,  ..., 7.56, 7.56, 7.56]
  gear_ratio: 1
  torque_control:
      VS_KP: -35730593.18196  # [-]
      VS_KI: -4499370.31680  # [s]
```

C

Rotor class changes

The following code is added into RAFT's Rotor class in order to include the frequency dependent nacelle velocity feedback term in the class. The intermediate code that is used as is, is not documented in this appendix. The full code can be found on RAFT's GitHub page.

```
##########################################################################################################
##########################################################################################################
######################### THIS CLASS IS COPIED FROM THE RAFT GITHUB PAGE ##################################
#### I'VE TRIED TO MAKE NO ADJUSTMENTS, BUT ALL I MADE ARE IN BETWEEN HASHTAGS LIKE THIS ####
## WITH TWO FULL ROWS IN THE TOP, FOLLOWED BY THEIR CODE WHICH I INACTIVATED, MY CODE, AND ##
################################### LASTLY TWO ROWS OF HASHTAGS ##########################################
##########################################################################################################
##########################################################################################################

# ... (omitting code in between)

##########################################################################################################
##########################################################################################################
        # self.k_float = 0 # np.zeros_like(self.Uhub), right now this is a single value, but this may change    <<< what is this?
        self.H_LP = np.ones_like(self.w)       # Low-pass transfer function
        self.H_N = np.ones_like(self.w)        # Notch transfer function
        self.k_float = np.zeros_like(self.w)   # Velocity feedback gain over a range of frequencies
##########################################################################################################
##########################################################################################################

# ... (omitting more code)

##########################################################################################################
##########################################################################################################
        # self.k_float = -turbine['pitch_control']['Fl_Kp']
        # self.k_float = np.zeros_like(self.w)  # Set to 0 since there seems to be a discrepancy with its definition

        # Get gain values from YAML-file
        self.Fl_Kp = turbine['pitch_control']['Fl_Kp']                  # Tower-top fore-aft motion's proportional gain
        Fl_Kp_cutoff_freq = turbine['pitch_control']['Fl_Kp_cutoff']    # 2nd-order low-pass filt. cut-off freq. [rad/s]
        Fl_Kp_damp_ratio = turbine['pitch_control']['Fl_Kp_damp_ratio'] # 2nd-order low-pass filt. damping ratio
        Fl_Kp_notch = turbine['pitch_control']['Fl_Kp_notch']           # Notch filter center [rad/s]
        Fl_Kp_Qfactor = turbine['pitch_control']['Fl_Kp_Qfactor']       # Notch factor Q factor

        omegas_notch = np.linspace(0, 5, 200)
        samp_freq = omegas_notch[len(omegas_notch)-1] + (omegas_notch[len(omegas_notch)-1] - omegas_notch[len(omegas_notch)-2]) / 2
        notch_freq = Fl_Kp_notch
```

```
        quality_factor = Fl_Kp_Qfactor
        b_notch, a_notch = signal.iirnotch(notch_freq, quality_factor, samp_freq*2)
        freq, h = signal.freqz(b_notch, a_notch, fs=samp_freq*2)
        # Calculate the magnitude of the transfer function
        H_N_transfer_function = np.abs(h)
        # Get values from method to match with omegas length and stepsize
        interpolator = interpolate.interp1d(freq, H_N_transfer_function)
        self.H_N = interpolator(self.w)
        # Determine transfer functions
        for i, w_i in enumerate(self.w):
            self.H_LP[i] = 1 / np.sqrt(1 + (w_i / Fl_Kp_cutoff_freq)**4)
        H_tot = self.H_LP * self.H_N
        self.k_float = self.Fl_Kp * H_tot

        # Create a plot
        plt.figure(figsize=(11.69, 8.27 / 1.5))
        plt.plot(self.w, H_tot, linestyle='solid', label='Total', linewidth=3.0)
        plt.plot(self.w, self.H_LP, linestyle='dashed', color='orange', label='Second-order
            low-pass')
        plt.plot(self.w, self.H_N, linestyle='dashed', color='green', label='Notch')
        plt.axvline(x=Fl_Kp_cutoff_freq, color='dimgrey', linestyle='dotted')
        plt.axvline(x=Fl_Kp_notch, color='dimgrey', linestyle='dotted')
        plt.text(Fl_Kp_cutoff_freq-0.025, -0.1, r'$\omega_{c}$', color='dimgrey')
        plt.text(Fl_Kp_notch-0.025, -0.1, r'$\omega_{ce}$', color='dimgrey')
        plt.xlabel('Frequency [rad/s]')
        plt.ylabel('Magnitude')
        plt.title('Transfer functions')
        plt.legend(loc='upper right')
        plt.grid(True)
        plt.tight_layout()
        plt.show()

        self.k_float = -np.array(self.k_float)  # Flipped sign convention for ROSCO
            controller
##########################################################################################################
##########################################################################################################

# ... (omitting more code)

##########################################################################################################
##########################################################################################################
            # NOTE: SINCE SELF.K_FLOAT WAS A CONSTANT THERE WAS NOTHING HERE AND THE MINUS
                SIGN WAS ALREADY DONE BEFORE!
            if Fl_Kp == 0:
                self.k_float = np.zeros_like(self.k_float)
            elif Fl_Kp == 1:
                self.k_float = self.k_float
##########################################################################################################
##########################################################################################################

# ... (omitting more code)

                # control transfer function
##########################################################################################################
##########################################################################################################
                # C[iw] = 1j * omega * (dQ_dU - self.k_float * dQ_dPi / self.Zhub) / D[iw]
                C[iw] = 1j * omega * (dQ_dU - self.k_float[iw] * dQ_dPi / self.Zhub) / D[iw]
##########################################################################################################
##########################################################################################################

# ... (omitting more code)

                # Complex aero damping
##########################################################################################################
##########################################################################################################
                # T = 1j * omega * (dT_dU - self.k_float * dT_dPi / self.Zhub) - ( E[iw] * C[
                    iw])
                T = 1j * omega * (dT_dU - self.k_float[iw] * dT_dPi / self.Zhub) - ( E[iw] *
                    C[iw])
##########################################################################################################
##########################################################################################################
```

D

Python functions

This part of the appendix shows some of the functions used during the methodology found in figure 5.1. More code is written for the full Python wrapper, but the function mentioned here are highlighted throughout chapter 5.

D.1. calcAerodynamics function

The following Python code is used to determine the frequency dependent aerodynamic contributions. It uses RAFT's Rotor class to do so, as well as the DLC dependent data and the previously determined static pitch angle.

```
def calcAerodynamics(designDir, staticPitch, DLC, V_mean, display=0, Fl_Kp=1):
    """
    Plots and calculates all individual contributions of the aerodynamic contributions vs the
        frequency.

    Input parameters
    ----------------
    designDir   : Design directory of the design that has to be evaluated, e.g. "M:Coupling\
        designs/15MW\VolturnUS"
    staticPitch = Pitch based on average wind speed [deg]
    DLC         = Design load case number, corresponding to information about the wave and
        wind conditions
    display     = Display mode: 0 = no plots, >1 = plots

    Returns
    -------
    F_aero0 = Steady aero forces and moments acting on COG (6x1)
    A_aero  = Frequency dependent aerodynamic added mass around COG (6x6)
    B_aero  = Frequency dependent aerodynamic damping around COG (6x6)
    F_aero  = Wind thrust force excitation spectrum acting on COG (6x1)
    Plots and xlsx data.
    """

    # Define inputs
    baseDir = 'N:\Jurrien\Coupling'
    design_input = os.path.join(designDir, 'excelInput.xlsx')
    yaml_input = os.path.join(designDir, 'turbine.yaml')
    with open(yaml_input) as file:
        yaml_file = yaml.safe_load(file)
    h5mInput = os.path.join(designDir, 'XML\\initial\\files_out\\readOutBEM.DIFFRAC.h5m')

    # Define output paths
    postProcessingDir = os.path.join(baseDir, 'postProcessing')
    xlsx_aero_path = os.path.join(postProcessingDir, 'aeroData.xlsx')

    # Load inputs needed for Rotor class initialisation
    # For turbine:
    turbineDir = yaml_file['turbine']
    # For w:
```

```
    # Opens the file for reading
    h5m = h5py.File(h5mInput, 'r')
    # Gets the body group which contains all the data, keys() can be used to see what the group contains
    body = h5m.get('Body_Nr_1')

    # Initialize wind speeds for plotting
    _, _, _, _, wind = loadCases(designDir, DLC)
    if V_mean == ["loadCase"]:
        V_mean = np.array([wind[0]])
        # Gets the frequency values used in DIFFRAC
        omegas = np.array(body.get('wave frequency'))
    else:
        V_mean = V_mean
        # Create frequencies for plotting aero contribution more neat
        omegas = np.linspace(0.01, 2, 200)
    freqs = np.array(omegas) / 2.0 / np.pi
    # Initialise the Rotor class from raft_rotor
    raftRotor = Rotor(turbineDir, omegas)
    # Initialise case dict
    case = dict(wind_speed='',
                wind_heading='',
                turbulence='',
                turbine_status='',
                yaw_misalign='')
    # Initialise arrays
    f_aero0 = np.zeros([len(V_mean), 6])
    f_aero = np.zeros([len(V_mean), len(omegas)], dtype=complex)
    b_aero = np.zeros([len(V_mean), len(omegas)])
    a_aero = np.zeros([len(V_mean), len(omegas)])
    V_w = np.zeros([len(V_mean), len(omegas)])
    c_exc = np.zeros([len(V_mean), len(omegas)], dtype=complex)
    # Initialize arrays
    A_aero = np.zeros([6, 6, len(omegas), len(V_mean)])
    B_aero = np.zeros([6, 6, len(omegas), len(V_mean)])
    F_aero = np.zeros([6, len(omegas), len(V_mean)], dtype=complex)
    F_aero0 = np.zeros([len(V_mean), 6])
    # Initialise excel for data storage (used in post processing)
    wb = openpyxl.Workbook()
    ws = wb.active
    # Convert staticPitch into array (in case of 1 DLC being considered, so indices can be used)
    if len(V_mean) == 1:
        staticPitch = np.array([staticPitch])
    else:
        staticPitch = np.array(staticPitch)

    # Create the case dict for three wind speeds
    for u in range(len(V_mean)):
        case[u] = dict(wind_speed=V_mean[u],
                       wind_heading=wind[1],
                       turbulence=wind[2],
                       turbine_status=wind[6], # Add;If wind speed is beyond operating limits : turbine_status!=operating
                       yaw_misalign=wind[1])

        # Use RAFT to run CCBlade simulation
        f_aero0[u], f_aero[u], a_aero[u], b_aero[u], V_w[u], c_exc[u] = raftRotor.calcAeroServoContributions(case[u], np.deg2rad(staticPitch[u]), display=display, Fl_Kp=Fl_Kp)
        # Writing to excel
        headers = [f'a_aeroU{V_mean[u]}', f'b_aeroU{V_mean[u]}']
        # Write the headers to the Excel file (only once)
        for col_num, header in enumerate(headers, start=1):
            ws.cell(row=1, column=col_num + (u * 2), value=header)
        # Write the data to the Excel file below the headers
        for row_offset, (a, b) in enumerate(zip(a_aero[u], b_aero[u]), start=1):
            ws.cell(row=row_offset + 1, column=col_num - 1 + (u * 2), value=a)
            ws.cell(row=row_offset + 1, column=col_num + (u * 2), value=b)

        # Hub reference frame relative to CoG
```

```
        hHub = yaml_file['turbine']['hHub']
        cog_z = pd.read_excel(design_input, sheet_name='XMLinput', usecols="D").iloc[0, 0]  # [m]
        T_floater = pd.read_excel(design_input, sheet_name='XMLinput', usecols="H").iloc[0, 0]  # [m]
        rHub = np.array([0, 0, hHub+abs(T_floater - cog_z)])  # [m] vector from CoG reference point to hub

        # Convert coefficients to CoG
        for i in range(len(omegas)):
            A_aero[:, :, i, u] = translateMatrix3to6DOF(np.diag([a_aero[u, i], 0, 0]), rHub)
            B_aero[:, :, i, u] = translateMatrix3to6DOF(np.diag([b_aero[u, i], 0, 0]), rHub)

        # Convert forces to CoG
        F_aero0[u] = transformForce(f_aero0[u], offset=rHub)  # Mean forces and moments

        for iw in range(len(omegas)):
            F_aero[:, iw, u] = translateForce3to6DOF(np.array([f_aero[u, iw], 0, 0]), rHub)

    # Write data to the xlsx file (used for postprocessing)
    wb.save(xlsx_aero_path)
```

D.2. addAeroToH5 function

The following function is used to include the frequency dependent aerodynamic contributions in the h5-file, that already contains the hydrodynamic contributions via BEM calculations, excluding the mooring contributions and the hydrodynamic viscous damping.

```
def addAeroToH5(designDir, h5file, DLC, A_aero, B_aero, F_aero):
    """
    Function to read the *.h5 file and find the additional added masses, damping terms and forces and moments. These are
    overwritten by the latest terms, determined by using RAFT, so DIFFRAC can rerun the calculations accordingly.

    Parameters
    ----------
    designDir   : design directory of the design that has to be evaluated, e.g. "M:Coupling\designs\\15MW\VolturnUS"
    A_aero      = Aerodynamic added masses [6x6], for each frequency, as determined by using RAFT
    B_aero      = Aerodynamic damping terms [6x6], for each frequency, as determined by using RAFT
    F_aero      = Aerodynamic forces and moments [1x6], for each frequency, as determined by using RAFT

    Returns
    -------
    Updated *.h5 file
    """
    # Find out what the wave direction is used in the DLC
    _, _, _, wave, _ = loadCases(designDir, DLC)
    waveDir = np.deg2rad(float(wave[1]))  # [rad]

    # Read and write the aerodynamic contributions to the *.h5 file
    with h5py.File(h5file, 'r+') as f:
        # Find the index of the group with the name 'WAVERESULT'
        j_group = None
        for i_group, group_name in enumerate(f.keys()):
            if group_name == 'WAVERESULT':
                j_group = i_group
                break
        # Find the index of the (closest) wave direction as used in the DLC, to write aerodynamic excitation at that dir
        if j_group is not None:
            group = list(f.keys())[j_group]
            waveDirs = f[group]['waveDir'][:]
            differences = np.abs(waveDirs - waveDir)
            waveDirIndex = np.argmin(differences)

```

```
        # Find aerodynamic entries and loop over the frequencies
        if j_group is not None:
            # Select the group which contains the frequency information
            # group = f[list(f.keys())[j_group]]
            group = list(f.keys())[j_group]
            # Initialize the omega array
            om = []
            # Collect the frequencies
            for i_om, om_group_name in enumerate(f[group]):
                # Filter the 'waterDepth', 'waveDir', 'waveFreq' and 'waveNumber' out of the list
                if om_group_name.startswith('OMEGA='):
                    # Collect the group names as strings
                    str = om_group_name
                    # Split a result (e.g.: 'OMEGA=0.6280') in two entries ('['OMEGA', '0.6280']')
                    om_str = str.split('=')
                    # Only store the actual omegas values
                    om.append(float(om_str[1]))

                    # Define the group key for the specific frequency to be opened
                    om_group = f[group][str]
                    # Read the current additional added terms used in the simulation for the given frequency
                    A_add = om_group['AMAS_ADD'][:]
                    B_add = om_group['BDAMP_ADD'][:]
                    # F_add_re = om_group['WAVEFORCES_ADD_r'][0, waveDirIndex][:]
                    # F_add_im = om_group['WAVEFORCES_ADD_i'][0, waveDirIndex][:]
                    F_add_re = om_group['WAVEFORCES_ADD_r'][:]
                    F_add_im = om_group['WAVEFORCES_ADD_i'][:]
                    # Initialise the new terms to overwrite the current ones
                    A_add_new = np.zeros_like(A_add)
                    B_add_new = np.zeros_like(B_add)
                    F_add_re_new = np.zeros_like(F_add_re)
                    F_add_im_new = np.zeros_like(F_add_im)
                    # Fill in the new matrices with the aerodynamic data retrieved using RAFT
                    for i in range(len(A_add)):
                        for j in range(len(A_add)):
                            F_add_re_new[0, waveDirIndex, i] = np.real(F_aero[i, i_om])  # TODO: Add way to include wind and wave misalignment
                            F_add_im_new[0, waveDirIndex, i] = np.imag(F_aero[i, i_om])  # TODO: Add way to include wind and wave misalignment
                            A_add_new[i, j] = A_aero[i, j, i_om]
                            B_add_new[i, j] = B_aero[i, j, i_om]
                    # Write the new matrices in the *.h5 file
                    om_group['AMAS_ADD'][:] = A_add_new
                    om_group['BDAMP_ADD'][:] = B_add_new
                    # om_group['WAVEFORCES_ADD_r'][0, waveDirIndex][:] = F_add_re_new
                    # om_group['WAVEFORCES_ADD_i'][0, waveDirIndex][:] = F_add_im_new
                    om_group['WAVEFORCES_ADD_r'][:] = F_add_re_new
                    om_group['WAVEFORCES_ADD_i'][:] = F_add_im_new
```

D.3. calcLinearViscousDamping function

The following Python code is used to determine the linear hydrodynamic viscous damping that can be used as an input in DIFFRAC. To do so, it uses the given quadratic viscous damping and linearizes this using the response based on the DLC conditions.

```
    def calcLinearViscousDamping(designDir, DLC, A_aero, B_aero, F_aero):
    """
    Creates a linear viscous matrix in a text-file format, based on the quadratic one and DLC conditions.

    Parameters
    ----------
    designDir   : design directory of the design that has to be evaluated, e.g. "M:Coupling\designs\\15MW\VolturnUS"
    DLC         = load case number, information about the wave and wind conditions
    A_aero      = Frequency dependent aerodynamic added mass around COG (6x6)
    B_aero      = Frequency dependent aerodynamic damping around COG (6x6)
```

```
11
12      Returns
13      -------
14      Text-file in designDir map containing Linear viscous damping
15      """
16
17      # Define input paths
18      quadraticViscDampTxt = os.path.join(designDir, 'quadraticViscousDampingFloater.txt')
19      design_input = os.path.join(designDir, 'excelInput.xlsx')
20
21      # Define output paths
22      linearViscDampTxt = os.path.join(designDir, 'linearViscousDampingFloater.txt')
23
24      # DIFFRAC directories
25      DIFFRACSUITEDir = r'C:\Users\jschipper\Documents\Software\DIFFRAC\New\Fortran_exe\current
            \DIFFRACSUITE\DIFFRACSUITE.exe'
26      initialRunDir = os.path.join(designDir, 'XML\initial')
27      BcritRunDir = os.path.join(designDir, 'XML\quadraticVisc')
28      BlinRunDir = os.path.join(designDir, 'XML\linearVisc')
29
30      # Copy outputs from initial BEM calculations into the other folders
31      folders_to_copy = ['files_out', 'files_tmp', 'files_vtk']
32      for folder_name in folders_to_copy:
33          source_folder = os.path.join(initialRunDir, folder_name)
34          destination_folder_quad = os.path.join(BcritRunDir, folder_name)
35          destination_folder_lin = os.path.join(BlinRunDir, folder_name)
36          shutil.copytree(source_folder, destination_folder_quad, dirs_exist_ok=True)
37          shutil.copytree(source_folder, destination_folder_lin, dirs_exist_ok=True)
38
39      # Add aerodynamic contributions to the *.h5-file
40      h5file = os.path.join(BcritRunDir, 'files_tmp\\readOutBEM.h5')
41      addAeroToH5(designDir, h5file, DLC, A_aero, B_aero, F_aero)
42
43      # Run DIFFRAC with the viscous damping as a percentage of critical damping
44      os.chdir(BcritRunDir)
45      call([DIFFRACSUITEDir, 'readByDIFFRACBcrit'])
46
47      # Load wave conditions for linearization
48      _, _, _, waves, _ = loadCases(designDir, DLC)
49      waveSpectrum = waves[2]
50      omegas = waves[4]
51      dw = omegas[1] - omegas[0]
52
53      # Load RAOs from the BcritRun
54      RAODB = pymarin.io.raodb.read(os.path.join(BcritRunDir, 'files_out\\output_motion.raodb.
            ini'))
55      # RAODB = pymarin.io.raodb.read(os.path.join(initialRunDir, 'files_out\\output_motion.
            raodb.ini'))
56      # Loop initialization
57      DOF = 6
58      velRAO = np.zeros([DOF, len(omegas)], dtype=complex)
59      velResp = np.zeros([DOF, len(omegas)])
60      velRMS = np.zeros(DOF)
61
62      # Load calculated headings
63      headings = RAODB.signals[3][:]  # [deg]
64      # Find index number of headings from which we want the RAOs
65      # Wanted headings
66      wantedHeadings = [0, 90]        # [deg]
67      # Indices
68      headingIndices = [np.where(headings == value)[0][0] for value in wantedHeadings]
69
70      # Find RAOs
71      # For 0 deg: surge, heave, pitch, and yaw
72      RAO_surge = RAODB.signals[5][:][0][0][0][headingIndices[0], :]
73      RAO_heave = RAODB.signals[7][:][0][0][0][headingIndices[0], :]
74      RAO_pitch = RAODB.signals[9][:][0][0][0][headingIndices[0], :]
75      RAO_yaw = RAODB.signals[10][:][0][0][0][headingIndices[0], :]
76      # For 90 deg: sway and roll
77      RAO_sway = RAODB.signals[6][:][0][0][0][headingIndices[1], :]
78      RAO_roll = RAODB.signals[8][:][0][0][0][headingIndices[1], :]
```

```
    # Store in list
    posRAO = [RAO_surge, RAO_sway, RAO_heave, RAO_roll, RAO_pitch, RAO_yaw]
    # Determine responses


    for i in range(DOF):
        # Determine velocity RAOs
        velRAO[i, :] = posRAO[i] * 1j * omegas
        # Determine velocity responses
        velResp[i, :] = abs(velRAO[i, :]) * waveSpectrum.psd
        # Determine RMS
        velRMS[i] = getRMS(velResp[i, :], dw)
    # Translate RMS from deg to rad for rotational modes
    for i in range(DOF - 3):
        velRMS[i + 3] = np.deg2rad(velRMS[i + 3])

    # Read quadratic matrix (from Literature)
    quadraticViscDamp = np.loadtxt(quadraticViscDampTxt) / 1000  # [kN]
    # Rewrite to be around COG instead of at SWL
    T_floater = pd.read_excel(design_input, sheet_name='XMLinput', usecols="H").iloc[0, 0]  # [m]
    cog_z = pd.read_excel(design_input, sheet_name='XMLinput', usecols="D").iloc[0, 0]      # [m]
    rVisc = [0, 0, T_floater - cog_z]  # [m] Vector that goes from where you want to be, to where the r point curr. is
    quadraticViscDampCOG = translateMatrix6to6DOF(quadraticViscDamp, rVisc)

    # Linearize quadratic response based on the DLC dependent RMS values
    linearViscDamp = np.zeros([6, 6])
    for i in range(6):
        for j in range(6):
            linearViscDamp[i, j] = quadraticViscDampCOG[i, j] * velRMS[i]

    # Write the matrix to txt-file
    np.savetxt(os.path.join(designDir, linearViscDampTxt), linearViscDamp)

    # Plot to check the wave spectrum
    # plt.plot(omegas, waveSpectrum.psd)
    # plt.xlabel('Omegas')
    # plt.ylabel('Wave Spectrum PSD')
    # plt.title('Wave Spectrum PSD vs. Omegas')
    # plt.show()
```

D.4. runLinearViscous function

The following code is used to run the DIFFRAC simulation including the aerodynamic contributions and the linearized viscous damping. The DIFFRAC run re-determines the wave drift loads for the situation with these contributions included, and writes a hyd-file containing all the data to be used in the aNySIM simulations.

```
    def runLinearViscous(designDir, DLC, A_aero, B_aero, F_aero):
    """
    Does a DIFFRAC run using the DLC dependent linearized viscous damping and the aerodynamic contributions, but without
    mooring stiffness included (this will be added in the aNySIM-fredom simulation for this case).

    Parameters
    ----------
    designDir   : design directory of the design that has to be evaluated, e.g. "M:Coupling\designs\\15MW\VolturnUS"
    A_aero      = Frequency dependent aerodynamic added mass around COG (6x6)
    B_aero      = Frequency dependent aerodynamic damping around COG (6x6)

    Returns
    -------
    Hydrodynamic database in the 'linearVisc' folder that will be used for mooring linearization
    """
```

```

    # DIFFRAC directories
    DIFFRACSUITEDir = r'C:\Users\jschipper\Documents\Software\DIFFRAC\New\Fortran_exe\current\DIFFRACSUITE\DIFFRACSUITE.exe'
    BlinRunDir = os.path.join(designDir, 'XML\linearVisc')
    BlinRunXML = os.path.join(BlinRunDir, 'readByDIFFRACBlin.input.xml')

    # Load linearized viscous damping
    linearViscDampTxt = os.path.join(designDir, 'linearViscousDampingFloater.txt')
    linearViscDamp = np.loadtxt(linearViscDampTxt)

    # Write the linear viscous damping diagonals to the *.xml-file
    # Parse the *.xml-file
    tree = ET.parse(BlinRunXML)
    # Define the root of the *.xml-file
    root = tree.getroot()
    # Write the diagonals on the *.xml-file
    for i in range(len(linearViscDamp)):
        root[4][1][0][1][i][3].text = str(linearViscDamp[i, i])
    # Save the updated *.xml-file
    tree.write(BlinRunXML, encoding='iso-8859-1')

    # Add aero contributions to the *.h5-file
    h5file = os.path.join(designDir, 'XML\\linearVisc\\files_tmp\\readOutBEM.h5')
    addAeroToH5(designDir, h5file, DLC, A_aero, B_aero, F_aero)  # Note that the A_aero and B_aero should be in order of 'kN', not 'N'!

    # Run DBRESP with the updated files including aero contributions and linearized viscous damping
    os.chdir(BlinRunDir)
    call([DIFFRACSUITEDir, 'readByDIFFRACBlin'])
```

D.5. Functions for the mooring linearization process

The following functions are used to linearize the mooring response. The first functions are used as input in the last function, ”mooringLinearization”, that actually performs the linearization.

```
# -------------- Some general functions needed for the linearization process -------------- #
# generalFunc
def strList2floatList(strList):
    return [float(x) for x in strList]

def floatList2strList(floatList):
    return [str(x) for x in floatList]

def list2str(List):
    return ', '.join(List)

def str2list(string):
    return string.split(', ')

# ------------------------------- Code for post processing ------------------------------- #
# postProcessing
def h5m2Array(path, motion, DOF):
    import warnings
    warnings.filterwarnings("ignore")

    varArray = [motion + '_' + str(x) for x in range(DOF)]  # desired variables

    res = pymarin.io.read_freqdb(path)

    omega = np.array(res['Body_Nr_0'].frequencies)
    PSD = np.zeros((len(omega), len(varArray)))
    for j in range(len(varArray)):
        PSD[:, j] = res['Body_Nr_0'][varArray[j]][0][0][:]

    return omega, PSD


```

```python
# ------------------------------ Code to replace some strings ---------------------------- #
# inPlaceChange
def inPlaceChange(path, old_string, new_string):
    """
    Replace strings in a file. Needed for mooring linearization.

    Input
    -----
    Filename      : String containing the filename wherein a part is going to be replaced
    Old_string    : String that is going to be replaced
    New_string    : String that replaces the old string

    Returns
    -------
    Filename containing the new input.
    """
    with open(path) as f:
        s = f.read()
        if old_string not in s:
            print(('Error: "{old_string}" not found in {path}').format(**locals()))
            return  # Exit the function if the old_string is not found

    with open(path, 'w') as f:
        s = s.replace(old_string, new_string)
        f.write(s)


# ---------------------------------- Start of the code ---------------------------------- #
# --- Some general inputs
# aNySIM environment
aNySIMenv = 'C:\\Users\\jschipper\\Documents\\Software\\aNySIM_XMF_v2023.02.3\\
    aNySIM_XMF_v2023.02.3\\setenv.cmd'

# Model parameters
DOF = 6

# --- Linearization
def checkLinearization(dV, designDir, evalNr='', Iter=0):
    # Define paths
    xmfPath = os.path.join(designDir, 'XMF')
    xmfResultPath = os.path.join(xmfPath, 'FredomResults\\responsePSD.h5m')

    # Initialize linNeeded
    linNeeded = False
    # Load data from freDom run
    omegas, velPSD = h5m2Array(xmfResultPath, 'velocity', DOF)  # PSD = [omegas, variables]
    # Get the variance from the PSD
    Vvar = integrate.simps(velPSD, omegas, axis=0)
    # Get the standard deviation from the variance
    Vstd = np.sqrt(Vvar)                                        # std = [variables]
    # Determine new velocity perturbations according to the method from C.H.Housseine, C.
        Monroy and G.deHauteclocque.
    dVcalc = np.sqrt(8 / np.pi) * Vstd
    # Check percentual difference between velocity inputs
    temp = (np.abs(dVcalc - dV) / dV) * 100
    temp[Vstd < 1e-4] = 0
    # If any of the newly calculated dV > 1% different to the old one, linNeeded=True
    perDiff = np.amax(temp)
    tol = 1  # [%]
    linNotCorrect = perDiff > tol

    print('----------------- perDiff = ', perDiff)

    # Create new dV based on the linearization step
    dV = dVcalc

    if linNotCorrect:
        linNeeded = True

    return linNeeded, dV

```

```
103
104 def setVelPertubation(designDir, dV):
105     # Set paths
106     xmfPath = os.path.join(designDir, 'XMF')
107     fredom = os.path.join(xmfPath, 'main_freDom_responseSolver.xmf')
108     fredomTemp = os.path.join(xmfPath, 'main_freDom_responseSolverTemp.xmf')
109     # Copy temp to the real freDom file
110     cp(fredomTemp, fredom)
111     # Convert dV to string
112     velPerStr = list2str(floatList2strList(dV))
113     # Update *.xmf-file with the string
114     inPlaceChange(fredom, '_velPer', velPerStr)
115
116
117 def linearize(dV, designDir, evalNr=''):
118     # Set paths
119     xmfPath = os.path.join(designDir, 'XMF')
120     path_to_ft = r"C:\Users\jschipper\Documents\Software\aNySIM_XMF_v2023.02.3\
        aNySIM_XMF_v2023.02.3\bin\win64"
121     command_with_path = os.path.join(path_to_ft, "ft")
122     command_with_path_and_args = [command_with_path, "-c", "main_freDom_responseSolver.xmf",
        "-E", "fredom"]
123
124     # Run linearizations
125     i=1
126     linNeeded = True
127     while linNeeded:
128         # Set the velocity perturbations in the *.xmf-file
129         setVelPertubation(designDir, dV)
130
131         # Run aNySIM-freDom
132         os.chdir(xmfPath)
133         subprocess.call(command_with_path_and_args)
134         # os.chdir(xmfPath)
135         # call([aNySIMenv, 'ft -c main_freDom_responseSolver.xmf -E fredom'])
136         # os.system('ft -c main_freDom_responseSolver.xmf -E freDom')
137
138         # Check if new evaluation is needed, if yes, perform it
139         linNeeded, dV = checkLinearization(dV, designDir, evalNr, i)
140         i+=1
141
142     return dV
143
144
145 # --- runLinearization
146 def mooringLinearization(designDir):
147     """
148     Does the linearization for the mooring using aNySIM-fredom and the linearization method
        as shown in Housseine, Monroy, and de Hautecluque
149
150     Parameters
151     ----------
152     designDir   : design directory of the design that has to be evaluated, e.g. "M:Coupling\
        designs\\15MW\VolturnUS"
153
154     Returns
155     -------
156     Final results in the response folder
157     """
158
159     # --- Define inputs
160     xmfPath = os.path.join(designDir, 'XMF')
161     xmfResultPath = os.path.join(xmfPath, 'FredomResults\\responsePSD.h5m')
162
163     # --- Set dir in aNySIM env
164     os.chdir(xmfPath)
165
166     # --- Initial velocity steps for linearization
167     dV = [0.01, 0.01, 0.01, 1e-5, 1e-5, 1e-5]
168
169     # Check Linearization - aNySIM is rerun when needed
```

```
170     linNeeded, dV = checkLinearization(dV, designDir)
171     if linNeeded:
172         print('Lin Needed')
173         dV = linearize(dV, designDir)
174 
175     # Load data from final freDom run
176     omegas, posPSD = h5m2Array(xmfResultPath, 'motion', DOF)  # PSD = [omegas, variables]
177     responseSurge = posPSD[:, 0]
178     responsePitch = posPSD[:, 4]
179     results = pymarin.io.read_freqdb(xmfResultPath)
180     responseTension = results['OutputSet_Nr_0']['TlcpFairlead1'][0][0][:]
181 
182     return omegas, responseSurge, responsePitch, responseTension
```

E

Power Spectral Densities

The tested design load cases (DLCs) are shown in this appendix in a larger size. The first DLC is including the lower frequency range with wind excitation, included in the time-domain model, but not in the created frequency-domain model. The other DLCs are a larger representation of figure 6.8, but including the OpenFAST lumped-mass results for the mooring line tension.

E.1. Design Load Case 1

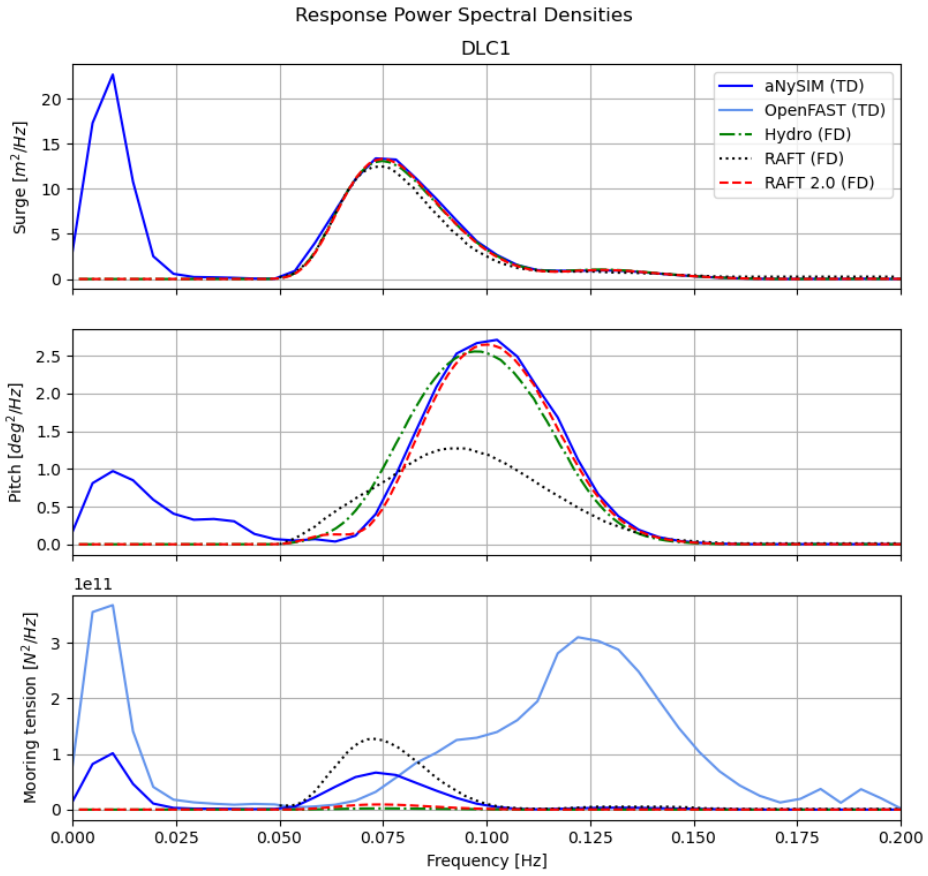

Figure E.1: Power spectral density response for surge, pitch and mooring tension, for DLC 1, showing the time-domain model results of aNySIM, as well as the frequency-domain model results from RAFT, RAFT 2.0 and the hydro approach

E.2. Design Load Case 2

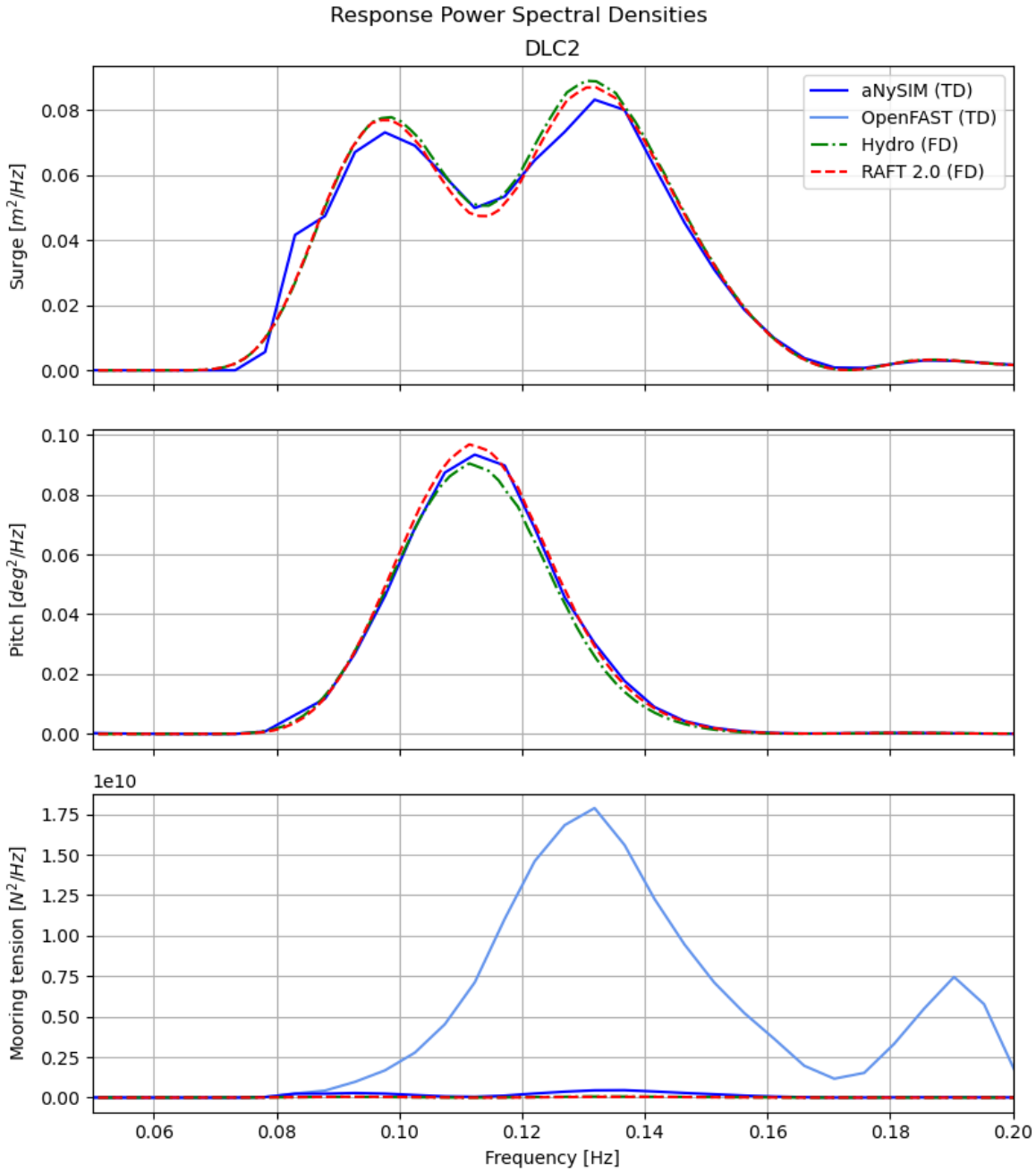

Figure E.2: Power spectral density response for surge, pitch and mooring tension, for DLC 2, showing the time-domain model results of aNySIM, as well as the frequency-domain model results from RAFT 2.0 and the hydro approach

E.3. Design Load Case 3

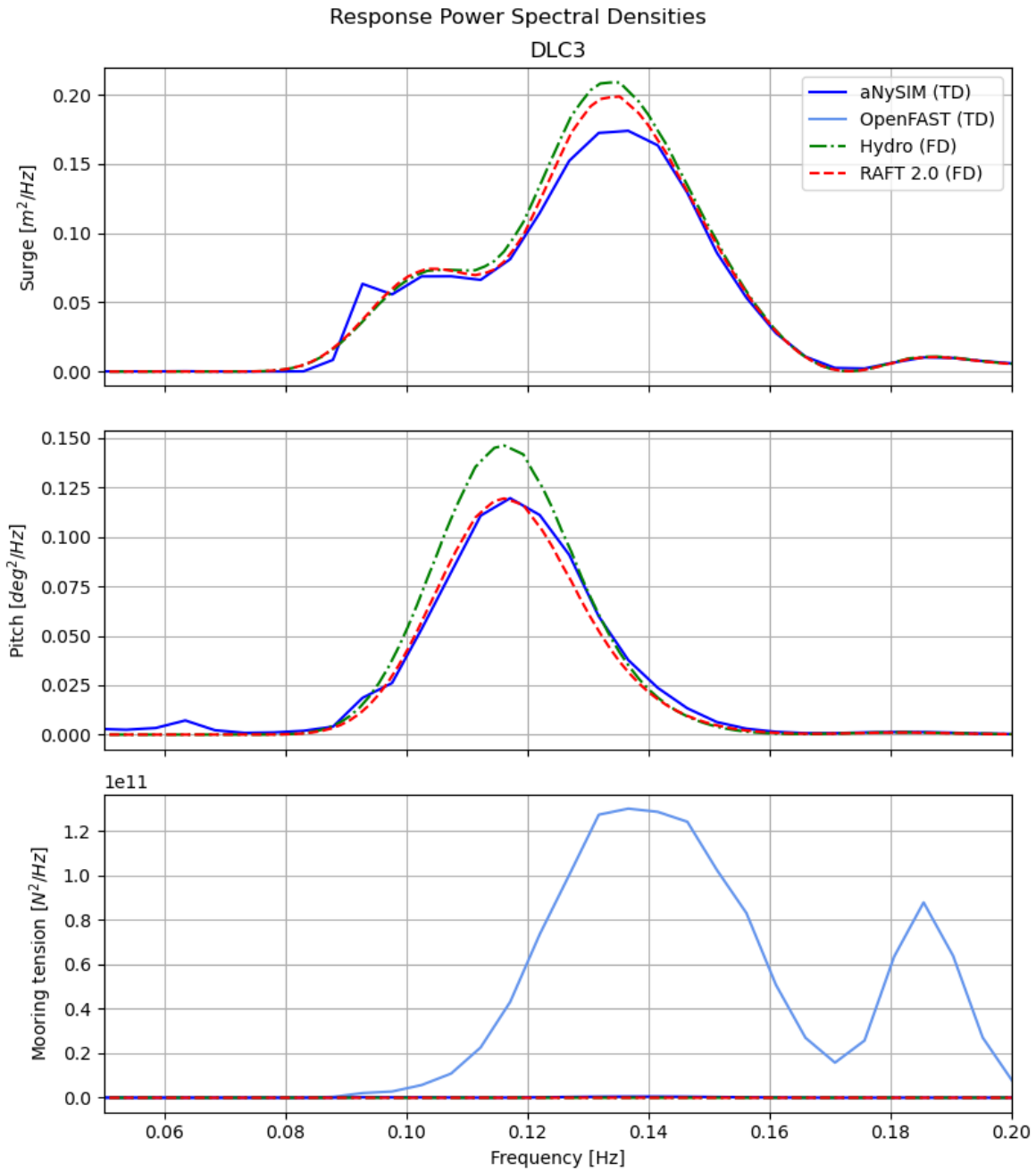

Figure E.3: Power spectral density response for surge, pitch and mooring tension, for DLC 3, showing the time-domain model results of aNySIM, as well as the frequency-domain model results from RAFT 2.0 and the hydro approach

E.4. Design Load Case 4

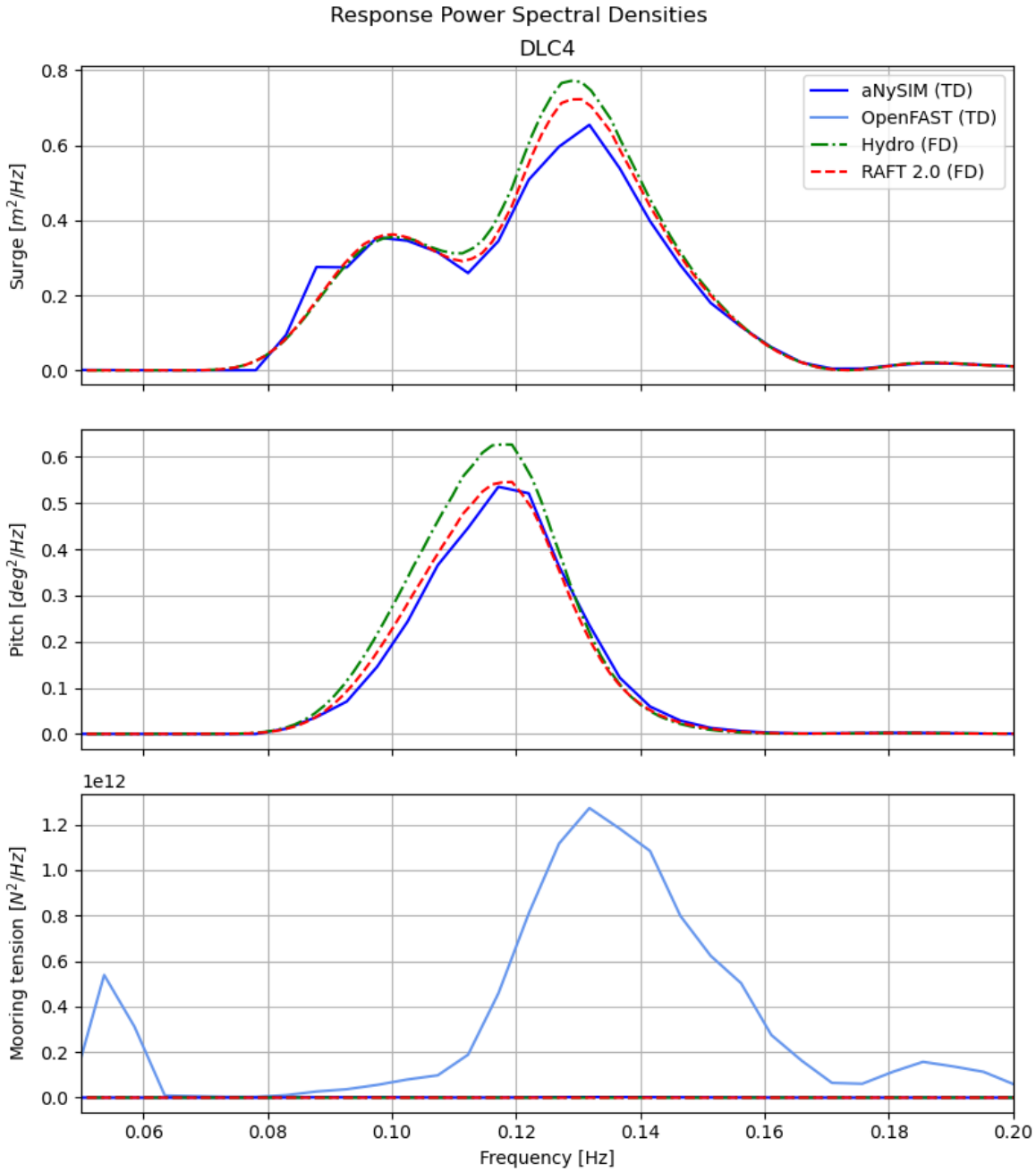

Figure E.4: Power spectral density response for surge, pitch and mooring tension, for DLC 4, showing the time-domain model results of aNySIM, as well as the frequency-domain model results from RAFT 2.0 and the hydro approach