Master of Science Thesis

Unsteady Aerodynamics of Floating Offshore Wind Turbines Under Surge Motion

A CFD Analysis with OpenFoam

George Spyridonos

April 7, 2023

Aerodynamics

TU Delft

Faculty of Aerospace Engineering

Delft University of Technology

Unsteady Aerodynamics of Floating Offshore Wind Turbines Under Surge Motion

A CFD Analysis with OpenFoam

Master of Science Thesis

For obtaining the degree of Master of Science in Aerospace Engineering
at Delft University of Technology

George Spyridonos

April 7, 2023

Faculty of Aerospace Engineering · Delft University of Technology

Delft University of Technology

DELFT UNIVERSITY OF TECHNOLOGY
DEPARTMENT OF AERODYNAMICS

The undersigned hereby certify that they have read and recommend to the Faculty of Aerospace Engineering for acceptance the thesis entitled **"Unsteady Aerodynamics of Floating Offshore Wind Turbines Under Surge Motion"** by **George Spyridonos** in fulfillment of the requirements for the degree of **Master of Science**.

Dated: April 7, 2023

Supervisors:

Dir.ir. Alexander Van Zuijlen

Dr. Wei Yu

Prof.dr.ir. Carlos Simão Ferreira

Abstract

Wind energy has witnessed rapid growth over the last 10 years and with the goal of the world's nations to reach climate neutrality in the near future it is expected to have an even bigger growth in the next decade. Due to its immense potential to harvest larger amounts of energy from the wind at a higher capacity factor, offshore wind energy is the key to reach the goal of a more sustainable future. The wind energy research community has been focusing all its efforts to bring the new concept of floating offshore wind turbines (FOWT) into the market. By installing the turbine on top of a floating platform the restrictions of offshore turbines regarding the sea depth are lifted and numerous new sites now become available for the wind energy market.

The floating platform of FOWT experiences several forces from the sea waves and, thus, imposes several motions on the turbine. Due to these motions several questions arise regarding the structural integrity, control efficiency and aerodynamic performance of this new wind turbine concept. Since aerodynamic models are utilized in almost every field involved in the design of a wind farm, its particularly important to make sure that the current industrial aerodynamic codes, like BEM, are able to capture the unsteady aerodynamics of this complex system. High-fidelity numerical codes and experiments are the only way to truly assess the performance of low-fidelity codes but also to gain further insight about the physics of this complex wind turbine concept.

Surge motion is recognized by all authors as one of the most important motions imposed to FOWT. Thus, in the present thesis the impact of surge motion in the aerodynamics of FOWT is investigated by employing a high-fidelity blade-resolved CFD model. The CFD model is implemented in OpenFOAM, an open-source CFD software, and is extensively validated against the high-fidelity experiment and CFD model of the UNAFLOW project. After the validation of the CFD model, two test cases are examined. One with a low surge frequency/amplitude that represents a mild sea state and one with an extreme surge frequency/amplitude which may lead to the creation of propeller or vortex ring state phenomena that render BEM invalid.

The results of the two test cases are utilized in two ways. In the first way, CFD velocity field data are used as an input for the Ferreira-Micallef induction computation model to acquire the induction field of the simulated rotor and compare it to the induction field predicted by momentum theory which is an integral part of BEM models. In the second way, CFD velocity and vorticity fields are used to produce streamline scenes and q-criterion isosurface

scenes with the aim to provide clarity on whether propeller and or vortex ring states occur during the operation of FOWT.

During the comparison of induction fields computed by Ferreira-Micallef model and momentum theory model, it became apparent that due to the 2D nature of both models it is not possible to draw any significant conclusions about the accuracy of momentum theory. Specifically, due to the existence of 3D radial flow during the operation of FOWT Ferreira-Micallef model is not reliable and can not be used as a basis to compute the error of momentum theory. Having said that, in the mid-span region of the blade and when the FOWT is operating in the mild surge conditions, 3D flow was not present and a comparison of induction fields between momentum theory and Ferreira-Micallef models yielded very similar results. However, this observation merely gives an indication that momentum theory can give reasonable results for a FOWT when 3D radial flow is not present. The only way to truly assess the performance of momentum theory in the induction field prediction of FOWT is to compare its results to an induction computation model that makes use of 3D CFD velocity data but also has a mechanism to account for 3D radial flow.

After analysing the streamline scenes and the q-criterion isosurface scenes for the extreme surge test case, two observations were made. First, even though C_T is acquiring negative values at some part of the surge period, the streamtube does not seem to experience any propeller state since it is expanding at all times. Second, in all the scenes that were produced there was no large vortex ring visible in the wake, thus there is no indication that vortex ring state occurs for a FOWT surging under these harsh surge conditions.

The present thesis contributes in the better understanding of FOWT aerodynamics and provides clarity on the ambiguous research topic of whether propeller/vortex ring states occur during their operation. Furthermore, the high-fidelity blade-resolved CFD model which was developed can be used as a starting point for several future research efforts.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Wind as a renewable energy source

The Paris agreement [1] formulated on 12 December 2015 by 196 countries signified a turn to a more sustainable future. By recognizing the imminent threat of global warming and the resulting climate change the participating parties agreed on several points which have as a main goal to battle global warming in the immediate future. As per article 2 of the agreement, a goal is set to limit the global temperature increase by less than 2 degrees Celsius over the next century. Furthermore, as per article 4 of the agreement, the parties have to take measures in order to accomplish a balance between greenhouse gases emissions and sinks. This so called "climate neutrality" can only be achieved if the CO_2 emissions experience a dramatic decrease in the near future. Therefore, turning towards carbon-free renewable energy sources is imperative for reaching the ambitious goals set by the Paris agreement.

Based on International Energy Agency (IEA), renewable energy statistics report [2], 13.5% of the total energy production for the year 2018 was attributed to renewable energy sources of which only 2% came from wind, solar, tidal and geothermal sources. Even though there is still a small contribution of wind energy to the global energy production, it is recognized as one of the most promising and fast-growing energy sources, reporting an annual growth of 23% in the period 1990- 2018. According to GWEC [3], in the year 2018 the total installed capacity was 591GW, with Europe, U.S. and China taking the lead. Moreover, GWEC also predicts that an annual growth of about 55GW will be accomplished until 2023 with the biggest percentage of the new installations coming from South-Eastern Asia and Latin America. Even though there is indeed a rapid growth in the market of wind energy, it is still far from reaching the levelized cost of energy (LCOE) of the common combined cycle power plants, with a common wind turbine LCOE ranging from 50 to 60 €/MWh and the LCOE of a common plant ranging from 35 to 50 €/MWh. Thus, it is necessary to reduce the cost of wind farms both by means of political actions and by further developing the current wind turbines/farms technology.

1.2 Offshore Wind Energy

As shown in Figure 1.1, the rapid growth of wind energy over the last 10 years is mostly attributed to the steady increase in installations of onshore wind farms. Until 2010, almost all the wind energy production originated from onshore farms, however, with the available inland sites continuously decreasing the alternative of offshore wind farms became more attractive. In fact, for the period 2010-2018 IEA reports that the number of installed offshore wind farms had an annual increase of about 30%.[4] Despite their steady yearly growth, the percentage of total offshore to total onshore installed capacity for the year 2018 was only 4%, admitting that the offshore wind farm market is still at an early developing stage. Offshore wind turbines exhibit some substantial benefits over inland turbines:

1. Capacity factor: simply put, capacity factor describes the average power output over the year relative to the maximum rated power capacity of an energy production system. Thus, it basically describes how often the system is operating at the maximum of its capabilities and could also be viewed as a measure of reliability of this system. Offshore wind turbines have the ability to achieve higher capacity factors than the onshore turbines. At the same time in some regions of the world, and depending the quality of wind, they are even able to match the capacity factors of coal/gas plants.[4]

2. Wind Turbine size: turbines size is continuously increasing. The bigger the size of the turbine the bigger the swept area and tip height and, thus, the bigger the rated power and capacity factor they can deliver.[4] In contrast with onshore turbines, offshore turbines can virtually make use of any turbine size since there is no limit in the size of the turbine that can be constructed in harbors and then transported in offshore sites.

3. Number of suitable sites: depending on the region of the world, there are numerous available offshore sites. Most of these sites also have advantageous wind conditions with large wind speeds and small turbulence intensity.

4. Minimal nuisance to the public: in contrast with onshore wind turbines, offshore turbines are installed far away from populated regions, thus there is neither noise nor visual nuisance to the public.

The abovementioned advantages make offshore wind turbines quite an attractive investment and constitute the main reason for their steady yearly 30% growth. However, even though they present such a potential, they still have a high cost when compared to common wind turbines, since their LCOE is ranging from 90 to 120 €/MWh. An offshore wind turbine design doesn't differ from an onshore turbine in terms of design. The basic difference of the offshore turbine is that it needs several extra actions to be taken in order to be resilient in the deep-sea environment. Among others these actions include a corrosion treatment for the turbine blades and a drying system for the nacelle/tower. Although these actions do increase the turbines LCOE, they are not the main reason for the high cost of the turbine. In fact, the biggest expense associated with the installation of the offshore turbines is the huge underwater structure of the turbine and mostly the foundation of this structure which lies on the seabed.

Together with this expense, another big contributor to the high LCOE is also the underwater cables that need to be installed for the connection of the offshore farm to the grid.

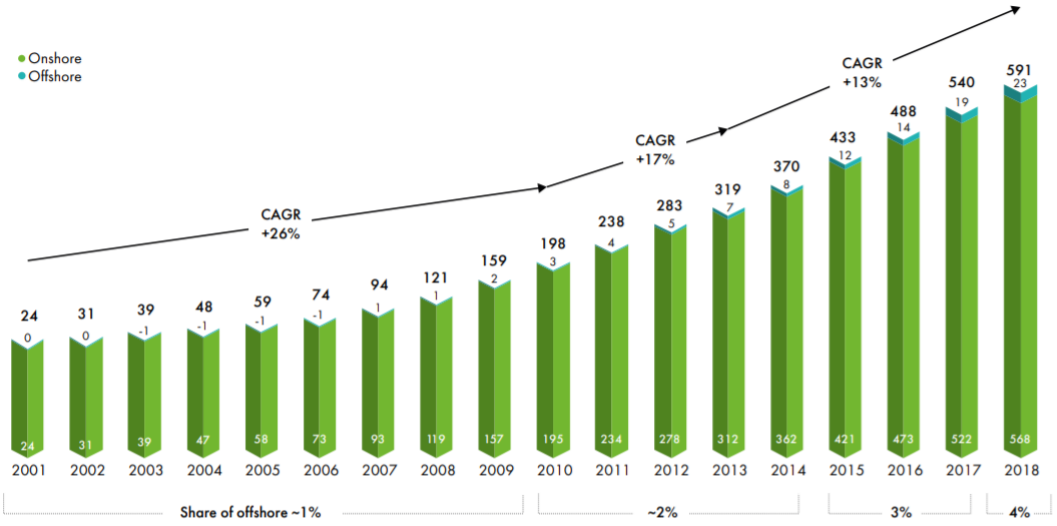


Figure 1.1: Development of installed onshore and offshore capacity over the years [3]

Having in mind the current high LCOE of offshore wind someone may wonder if the effort for making this technology more attractive/affordable is worthwhile. To better understand the immense potential of offshore wind energy the latest predictions of IEA [4] are presented in Figure 1.2. Europe alone is predicted to have an annual potential of 33840TWh which is more than enough to cover the current total global electricity demand. Thus, the inevitable conclusion is that in order to fully unlock the potential of wind energy, offshore wind has to be exploited.

An important setback for the exploitation of offshore wind energy is the fact that most of the available offshore sites in the world are characterized by a high sea depth, as shown in Figure 1.3. At the same time, the cost of installing and maintaining traditional offshore wind turbines in sites with high sea depths over 50 meters is prohibitive due to the need of a very large underwater substructure. In fact, in order to avoid a very high LCOE almost all of the offshore wind turbines (OWT) currently installed are only limited to sites with shallow waters with a depth of 40 to 50 meters. Therefore, one can easily observe that an important step to really unlock the immense potential of offshore wind energy is to reduce the LCOE of OWT for the high-depth sea sites. A promising idea towards this goal is the concept of floating offshore wind turbines.

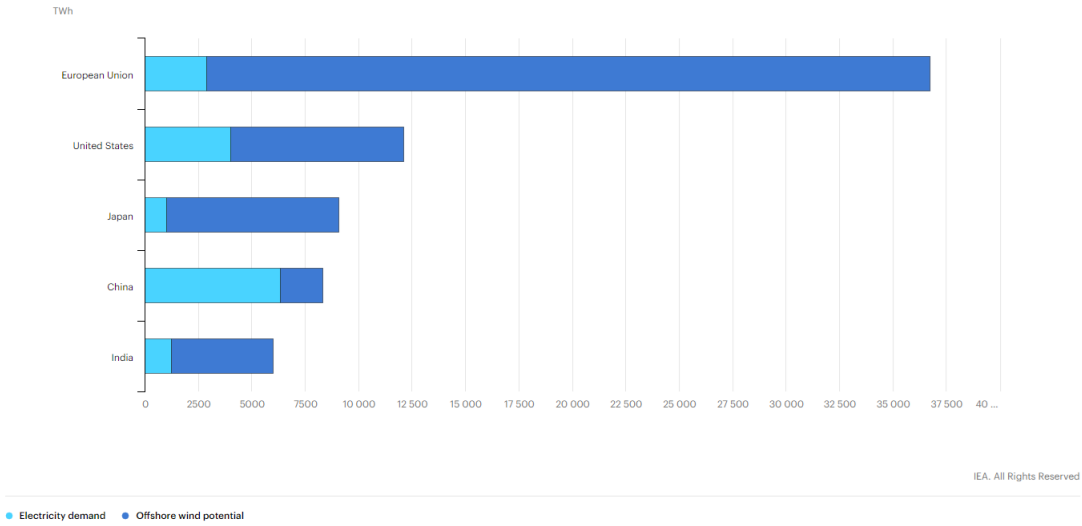


Figure 1.2: Offshore wind potential and the respective electricity demand for several key players in energy production. [4]

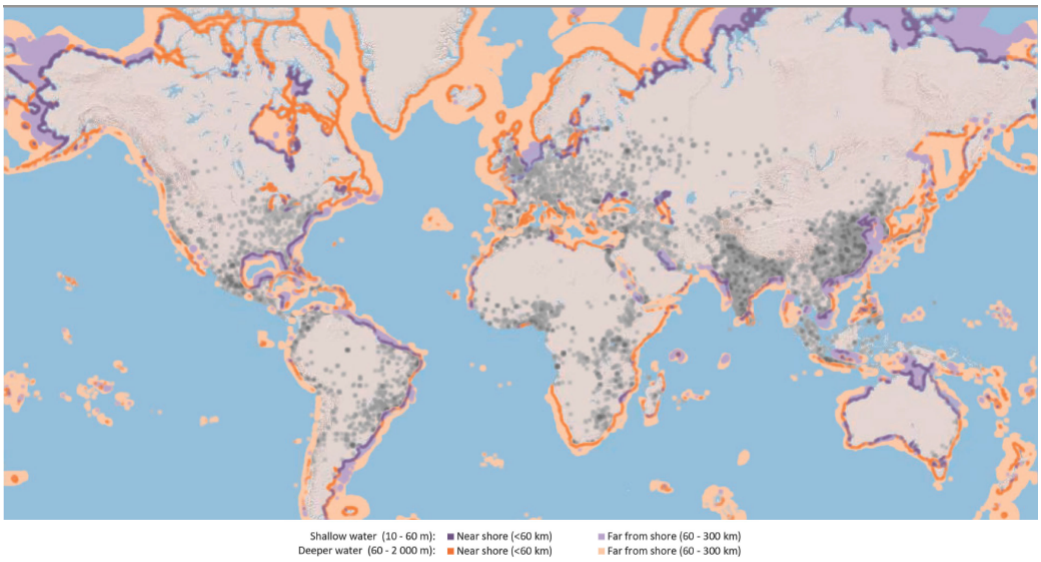


Figure 1.3: Global offshore sites map. [4]

1.3 Floating offshore wind turbines (FOWT)

As discussed in section 1.2, one of the major expenses of offshore wind turbines is the huge underwater structure and more importantly the foundation that has to be installed on the potential site's seabed. The high expense of this substructure is one of the main obstacles that prevent the exploitation of the huge wind energy potential of OWT. In fact, almost all

the OWT currently installed are only limited to sites with shallow waters with a depth of 40 to 50 meters due to the overwhelming cost of the substructure for high depths. [4] To overcome this limitation and fully unlock the potential of offshore wind energy, substantial research is currently performed in the field of floating offshore wind turbines (FOWT).

Floating turbines basically substitute the substructure and the seabed foundation with a floating platform which is anchored to the seabed. Without the need for an underwater structure, not only does the LCOE has the potential to decrease substantially but there is virtually no limitation in the sea depth of the possible offshore site. Removing this limitation is a substantial breakthrough, since it makes it possible to install OWT in numerous new sites and get into several new markets. Apart from cost reduction, floating turbines are also more friendly to the marine environment than the common OWT since they lack the seabed foundation.

Figure 1.4: Illustration of the three basic types of floating platforms for FOWT. Source: Josh Bauer, NREL.

Recognizing the potential of FOWT, several companies have already tried to test the concept of a floating wind turbine. Equinor was one of the first companies to test their Hywind floater design by installing the Hywind demo 2.3MW turbine offshore of Norway at a depth of around 220 meters. The Hywind floater design is basically a vertical ballast which is tethered to the seabed resembling the function of a spar buoy, as shown in Figure 1.4(left). After successfully operating this demo turbine, Equinor was also the first to install a floating offshore wind farm in 2017. The wind farm was installed offshore of Scotland and consisted of six 5MW turbines with a diameter of 154m and a Hywind floater platform. The farm showed a remarkable capacity factor of over 50% staying in line with the predictions of high capacity factors of offshore wind farms. Besides Equinor, Principle Power Inc. was also another company to develop their own floater concept for FOWT. The WindFloat floater consists of a ballast in

the center and 3 semi-submersible legs with a water-entrapment plate around the ballast, as shown in Figure 1.4(middle). The main advantage of such a design is that it can discharge the water trapped in entrapment plates in order to reduce the draft of the platform. This unique characteristic makes it possible to assemble the FOWT in the port and then transport the floating turbine to the site without making use of any crane vessels. Furthermore, besides the obvious reduction in the complexity and cost of the installation process, the WindFloat design also reduces the maintenance costs since it is possible to transport the turbine back to the port whenever it's needed.

Even though the current installed capacity of FOWT is only somewhere around 100-200MW, an abundance of new projects is scheduled for the coming years. In Europe, France, Scotland and Portugal have the most announced FOWT projects while in Asia, Japan has also announced a lot of FOWT projects. Since the technology of FOWT is still immature, most of the scheduled projects are for small wind farms of around 30MW and serve as a pilot for the market. Inevitably, the success of these projects is very crucial for the future of floating wind turbines. Even though a steady rise of FOWT is expected until 2030, with approximately 10% of the installed offshore farms predicted to be floating turbines [4], it is still very soon to conclude whether FOWT will be able to compete with the common WT or OWT.

One of the major concerns about the floating wind turbines is the fact that it is still obscure how these floating systems operate both hydrodynamically and aerodynamically. In fact, quite a lot of research is needed to determine the dynamics of these complex systems.

1.3.1 Importance of accurate description of the FOWT aerodynamic phenomena

Accurate predictions of the aerodynamic phenomena taking place in a wind turbine is crucial not only for floating offshore wind turbines but for WT in general. Aerodynamic models are used in almost every field involved in the design of a wind farm. A cost estimate analysis employs high-fidelity structural models which in turn take as input the aerodynamic forces predicted by aerodynamic models. High-fidelity aerodynamic models are also very important for the controllers regulating the operation of a wind farm. Controllers use the aerodynamic model's predictions to efficiently regulate yaw angle, blade pitch and rotation speed of the turbines for maximum power output while limiting the structural loads on the turbines. Besides aerodynamic models there are various other models involved in the design process of wind turbines, e.g. meteorological models. Decreasing the errors/uncertainties of any of the involved models leads to a more efficient end- result and thus a decreased overall LCOE. However, since aerodynamic models are involved in almost all the different parts of the design process, it is particularly important to increase their accuracy since they will also increase the accuracy of the various other models in play.

In the case of floating wind turbines there is an increased unsteadiness in the turbine aerodynamics due to the motions imposed to the rotor by the floating platform. The current state-of-the art research on WT aerodynamics [5] indicates that the accuracy of the present aerodynamic models are not accurate enough when it comes to unsteady phenomena. Therefore, further research on the unsteady aerodynamics of FOWT is of paramount importance

to unlock its full potential.

1.3.2 Motions of FOWT floating platforms

In contrast with common OWT which are fixed to the seabed with a rigid structure, floating offshore wind turbines undergo several motions during their operation. Both the sea waves and the wind impose some motions on the floating platform. These motions can be decomposed in six fundamental motions (6 degrees of freedom), as shown in Figure 1.5. Specifically, in an XYZ cartesian system defined on the platform, the six degrees of freedom consist of three rotational (roll, pitch and yaw) and three translational (heave, surge and sway) motions.

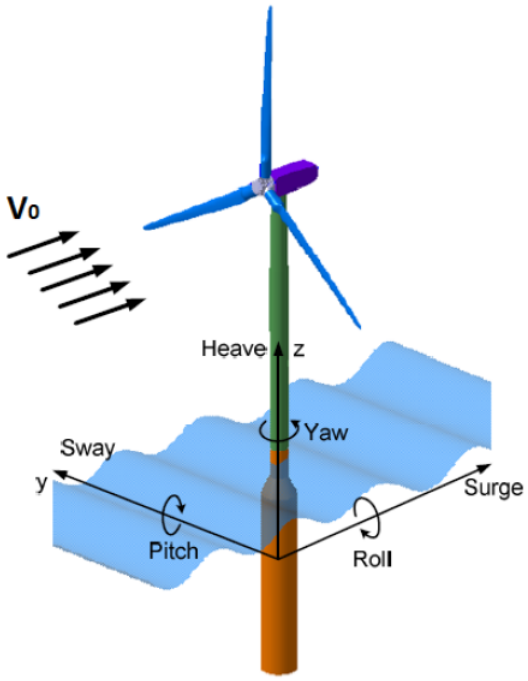


Figure 1.5: The six degrees of freedom of a FOWT. Picture found in [6].

The six additional DoFs that are present in the operation of a floating wind turbine introduce extra complexities and uncertainties when analyzing and designing these systems. Since these motions are not present in inland WT, the usual control strategies are not suitable for FOWT. In fact, dangerous resonances may even be created if the common control strategies are employed for these systems [7]. Besides the effect of these new motions to the control strategies of the turbine, there is also an effect on the aerodynamic and structural performance of the system. The motions imposed by the floater to the rotor may cause significant aerodynamic unsteadiness, due to dynamic inflow phenomena. Furthermore, since the turbine is subjected to various frequent motions during its lifetime, fatigue performance and the overall structural integrity is affected by new bending moments and tensions introduced into the system. Therefore, it is imperative to identify which of the 6 DoFs are the most important and how

do they specifically affect the power output, stability and structural integrity of FOWT.

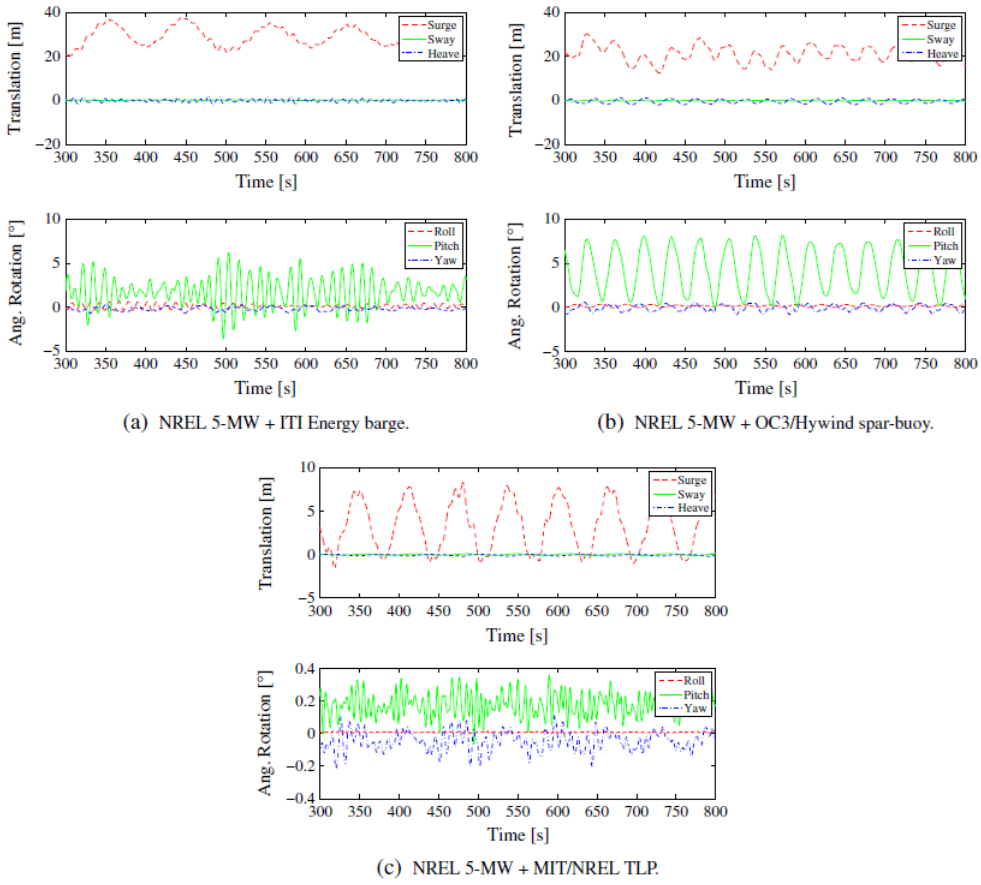


Figure 1.6: Kinematics of barge (a), spar-buoy (b) and tension leg (c) floating platforms when operating under rated conditions. [8]

As discussed in previous sections, there are various designs of floating platforms in the market. Despite the plurality of the designs, all the floating platforms can be classified in three basic types based on the way they achieve stability, as shown in Figure 1.4. Some platforms rely on a ballast (Figure 1.4 (left)), some on the mooring lines (Figure 1.4 (right)), and others on buoyancy forces (Figure 1.4 (middle)). Extensive research has been done in order to quantify the importance of each degree of freedom for each type of floating platform. Sebastian and Lackner [8] analyzed the kinematics of the three basic types of floaters using the aerohydrodynamic code FAST developed by NREL. Their work showed that the dominant translations/rotations of the floating system depend on the type of the floater, as shown in Figure 1.6. The barge platform, which is of similar design to the semisubmersible platforms, was mostly subjected to large pitch and surge motions when operating in rated conditions. Similarly, the spar-buoy platform exhibited surge motions of similar amplitude and pitch motions with slightly larger amplitudes. Lastly, the TLP (tension leg platform) design was again dominated by surge motions while pitch and yaw rotations of very small amplitude were also present. In all three types of platforms, pitch and surge were the most excited

motions out of the 6 DoFs. However, owing to the different stabilization technique of each platform, the amplitudes and frequencies of these two motions are different for each floater. Further research on the same field has been performed by Robertson, Jonkman and Matha [9, 10, 11, 12] who subjected the platforms in various design load case (DLC) scenarios and quantified the most dominant motion for each type of floater. Specifically, tension-leg and semi-submersible platforms were mostly dominated by low-frequency surge motions, while spar-buoy platforms were found to be more sensitive to pitch motions.

1.4 Thesis Perspective

As discussed in the previous sections, there is an outstanding potential in harvesting energy with floating offshore wind turbines (FOWT). Several researchers are currently focusing their efforts to answer the critical questions that this new concept introduces into the market. As discussed in section 1.3, because of the motions experienced by the FOWT, questions arise regarding the structural integrity, the control of such turbines and the aerodynamic performance of the turbine. Since aerodynamic models are involved in almost all the different parts of the design process, it is particularly important to make sure that the currently used aerodynamic models are accurate enough to describe the aerodynamics of this system.

Most industrial design codes employ BEM as the primary aerodynamic model. However, whether BEM is able to capture the unsteady aerodynamics of FOWT is still unclear. Furthermore, on harsh sea wave states where FOWT may experience extreme surge conditions, it is supported by some authors that 3D flow phenomena like propeller and vortex ring state may occur rendering BEM invalid. Several research efforts have been performed on these subjects and the conclusions are somewhat ambiguous. The availability of high-fidelity codes and wind tunnel measurements is of paramount importance in order to validate the codes of lower fidelity like BEM but also to gain further insight on the aerodynamic phenomena taking place in such a complex system.

In the present thesis, a high-fidelity blade-resolved CFD solver will be employed to analyze the aerodynamic performance of a FOWT under surge motion. The data produced by the CFD code will be used to assess the validity of momentum theory for a mild and an intense surge condition. Furthermore, the CFD data will be visualized in Paraview post-processing software to provide essential information about the 3D flow phenomena occuring during the operation of a surging rotor.

1.4.1 Research Objectives

The research objectives of the present thesis are as follows:

1. Build a high-fidelity blade-resolved CFD model in OpenFoam and validate it with high-fidelity experimental data.

2. Answer whether momentum theory is able to capture the induction field of a FOWT under surge motion.
3. Provide clarity on whether propeller and vortex ring (VRS) states take place during the operation of the FOWT under high surge frequencies/amplitudes.

1.4.2 Thesis Outline

The present work is structured in seven chapters. First, the literature review is presented in chapter 2. Second, the thesis project plan is explained in chapter 3. Next, the methodology that was followed is presented in chapter 4 and the model validation process is discussed in chapter 5. Finally, the results are presented in chapter 6 and the conclusion is given in chapter 7.

Chapter 2

Literature Review

In this chapter, the literature review of the present research topic is presented. In sections 2.1 and 2.2, the theoretical background of Blade Element Momentum (BEM) and Computational Fluid Dynamics (CFD) are presented, respectively. In section 2.3, some important unsteady aerodynamic phenomena that occur during the operation of FOWT are discussed. Next, in sections 2.4, 2.5 and 2.6, the current FOWT research efforts on BEM, CFD and other models of intermediate accuracy are presented, respectively. Lastly, in section 2.7, an overview of the surge conditions that have been already examined in the literature is given.

2.1 Blade Element Momentum (BEM) Theory

Owing to its simplicity and low computational cost, blade element momentum theory (BEM) is currently the most widely used aerodynamic model in WT design codes. BEM theory was first theorized and formulated by Glauert in 1935 and has been around for quite some years now. In its basis BEM is practically a combination of blade element theory and momentum theory. Blade element theory models the blade aerodynamics by breaking down the blade in sections and correlating them with the 2D aerodynamics of the respective airfoil used in these sections. On the other hand, momentum theory models the induction field of the WT using simplified versions of the conservation, momentum and energy equations.

2.1.1 Formulation of BEM theory equations

2.1.1.1 Axial Momentum Theory

Momentum theory is subdivided to axial and angular momentum theory based on the direction in which the conservation laws are applied. Axial momentum theory is basically the application of conservation laws in a 1D axial streamtube enclosing a WT which is represented

by an actuator disc, as shown in Figure 2.1. The actuator disc models the WT as a disc which exerts a uniform axial force on the flow field and can also be viewed as the equivalent of a rotating WT with an infinite number of blades.

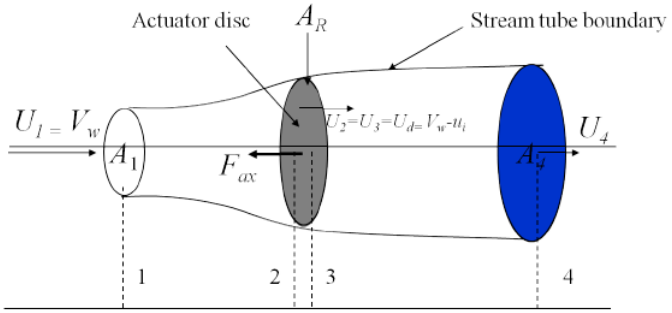


Figure 2.1: Stream tube employed for the application of axial momentum theory.[13]

The conservation laws are applied in four positions of the stream tube shown in Figure 2.1, one far upwind from the actuator (1), one exactly before the actuator (2), one exactly after (3) and one far downwind (4). Position 1 has a velocity equal to the incoming wind velocity and atmospheric pressure. Positions 2 and 3 are considered to be at an infinitesimally small distance to each other, thus they both have the same area $A_2 = A_3 = A_d$ and the same velocity $U_2 = U_3 = U_d$. Also, the axial force exerted on the flow by the actuator disc is the origin of a large pressure drop between positions 2 and 3. Finally, position 4 is the position where the flow has managed to recover back to atmospheric pressure. By applying the conservation laws on these four points the goal is find a relation between the axial force F_{ax} and the induced velocity u_i, i.e. the velocity reduction that is applied exactly before the disc due to the extraction of energy from the flow. First, conservation of mass is applied and the following relation is derived:

$$\dot{m} = \rho U_1 A_1 = \rho U_d A_d = \rho U_4 A_4 \quad , \tag{2.1}$$

then the conservation of momentum is applied between points 1 and 4:

$$\dot{m}(U_4 - U_1) = F_{ax} \quad , \tag{2.2}$$

and lastly conservation of energy is expresses as follows:

$$\dot{m}(0.5U_1^2 - 0.5U_4^2) = F_{ax}U_d = P_d \quad . \tag{2.3}$$

Connecting Eq. 2.1, 2.2 and 2.3 gives us the following relation for the velocity at the actuator:

$$U_d = 0.5(U_1 + U_4) = V_w(1 - \alpha) \quad , \tag{2.4}$$

where α is the axial induction factor and is defined as $\alpha = \frac{u_i}{V_w}$. Thus, combining all the above equations the relation between the axial force coefficient can be derived:

$$C_{F_{ax}} = C_T = \frac{F_{ax}}{0.5\rho V_w^2 A_d} = 4\alpha(1 - \alpha) \quad , \tag{2.5}$$

while the power coefficient can be defined is as follows:

$$C_P = \frac{P}{0.5\rho V_w^3 A_d} = 4\alpha(1 - \alpha)^2 \quad . \tag{2.6}$$

2.1.1.2 Angular momentum theory

Similarly to the axial momentum theory, the goal of angular momentum theory is to derive an equation connecting the tangential induced velocity and the torque exerted by the actuator disc to the flow. The stream tube concept is employed once more and three positions are taken, as shown in Figure 2.2. Positions 2 and 3 lie exactly before and after the actuator disc while position d lies exactly on the disc. The conservation laws are applied this time in the angular direction and the following equation is acquired for the torque exerted by the disc:

$$Q = \rho V_w(1 - \alpha)\omega_3 r^2 A_d \quad , \tag{2.7}$$

where ω_3 is the wake rotation speed. Based on the work of de Vries [14] the tangential induced velocity can be acquired through the following relation:

$$\omega_d = \frac{\omega_3}{2} = \alpha'\Omega \quad , \tag{2.8}$$

where Ω is the rotational speed of the rotor and α' is the tangential induction factor. Combining Eq. 2.7 and Eq. 2.8, the first can be rewritten as follows:

$$Q = \rho V_w(1 - \alpha)2\alpha'\Omega r^2 A_d \quad . \tag{2.9}$$

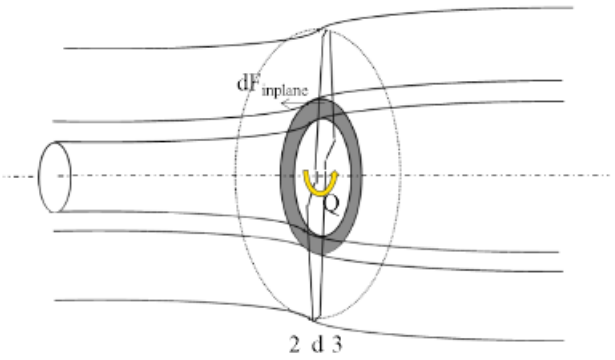


Figure 2.2: Stream tube and the positions taken for the application of angular momentum theory.[13]

2.1.1.3 Blade Elementh Theory

In the previous subsections the basic equations of the momentum theory were presented and relations between the induced velocities and the force/torque of the actuator have been established. The next step towards applying the BEM theory is to divide the stream tube of Figure 2.1 in several independent annuli, as shown in Figure 2.3. The axial and angular momentum theory equations are now applied in each of these annuli of radial length dr . Each annulus passes through the actuator disc forming a region like the highlighted region shown in Figure 2.3. The premise of blade element momentum theory is that the force of this ring region is the sum of the forces of all blade elements residing in this ring. Thus, by breaking down the blades in several radial sections of equivalent radial distances dr we can associate the forces exerted by the blade sections to the total average force exerted from the actuator ring to the stream tube annulus. The forces exerted by each blade section are approximated with the use of $2D$ airfoil data (C_l , C_d) of the airfoils used in these blade sections.

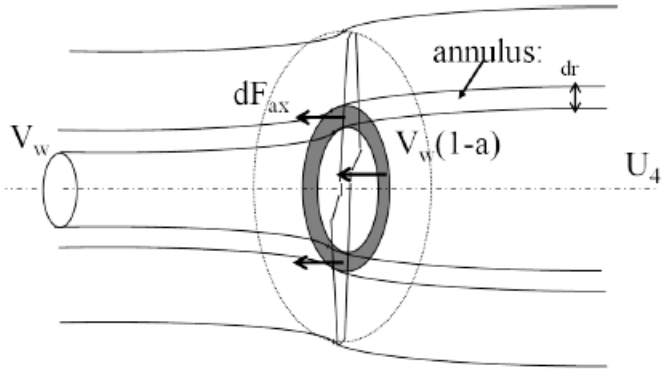


Figure 2.3: Division of the stream tube in several independent annulus. [13]

By using the velocities triangle shown in Figure 2.4, the axial force dF_{ax} is computed for each

blade section as follows:

$$dF_{ax} = L\cos\phi dr \quad , \tag{2.10}$$

where ϕ is the inflow angle and is defined as follows:

$$\phi = \arctan\left(\frac{1-\alpha}{\lambda_r(1+\alpha')}\right) \quad , \tag{2.11}$$

where λ_r is the local tip speed ratio and α' the tangential induction factor. It should be noted that the axial force computed in Eq. 2.10 has a small contribution from the drag force which is omitted. In fact, in most codes using BEM as their aerodynamic model the drag component is almost always omitted.

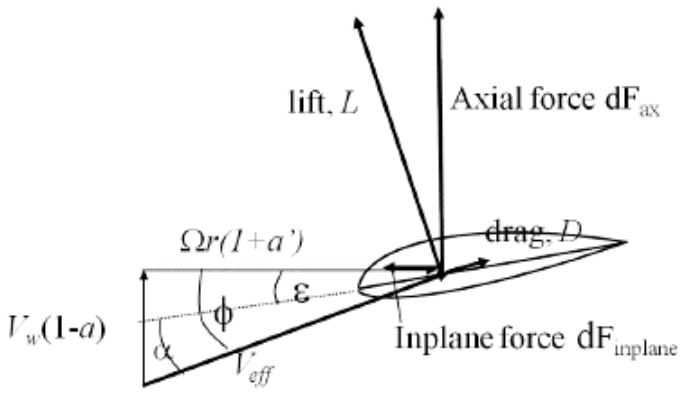


Figure 2.4: Velocity triangle for the airfoil of a blade element. Both axial and inplane forces are shown. [13]

Similarly, the tangential force and most importantly the torque exerted by the blade element can be approximated with the following relation:

$$dQ = dF_{inplane}r = L\sin\phi r \quad , \tag{2.12}$$

where the drag force is again considered to have minor contribution and is omitted.

2.1.1.4 BEM equations

Angular and axial momentum theory in conjunction with blade element theory yields the blade element momentum theory equations. Specifically, combining Eq. 2.5 with 2.10 and Eq. 2.9 with 2.12 gives us the following equations:

$$4\alpha V_w^2(1-\alpha) = V_{eff}^2 \sigma C_l \cos\phi \quad , \tag{2.13}$$

$$4\alpha' V_w(1-\alpha)\Omega r = V_{eff}^2 \sigma C_l \sin\phi \quad , \tag{2.14}$$

where ϕ is given by Eq. 2.11 and $_{eff}$ is computed as follows:

$$V_{eff}^2 = V_w^2\left(\lambda_r^2(1+\alpha')^2 + (1-\alpha)^2\right) \quad . \tag{2.15}$$

The lift coefficient of Eq. 2.13 and 2.14 are only a function of the angle of attack, thus the angle of attack in each blade section determines the magnitude of the forces on the right hand side of these equations. However, the angle of attack depends on the direction of V_{eff} which in turn depends on the value of the yet unknown induction factors, thus the only way to solve these equations is iteratively. The solution of this system of equations finally gives us the spanwise distribution of the induction factors as well as the spanwise distribution of thrust and toque on the rotor.

2.1.2 Assumptions and uncertainties of BEM theory

In the previous subsection the derivation of the blade element momentum equations was presented. The simplicity of these equations as well as the ability to solve them at the expense of a low computational time/cost explains why BEM is up to this day the most used aerodynamic model in wind turbine design codes. However, the simplicity and computational inexpensiveness comes with a sacrifice in the accuracy of the model. In fact, several assumptions and simplifications were made to arrive in Eq. 2.13 and 2.14. Each of these assumptions increases the error of the final solution and restricts the model's usability. In order to overcome these weaknesses of BEM, several add-ons were proposed over the years to strengthen the accuracy of the model. In the present section the assumptions/simplifications are presented and any proposed add-ons to resolve the underlying error is discussed.

2.1.2.1 Assumption of inviscid flow

The effects of viscosity are only considered in the blade element theory part of BEM and not in the momentum theory. In fact, in the case of blade element theory, the drag coefficient is taken from wind tunnel measurements which include the viscous effects and their contribution to drag. In momentum theory, the flow is assumed to be inviscid and any viscous phenomena are neglected. Even though it is difficult to assess whether viscosity effects are important for the accuracy of the method, they are generally considered to have a minor effect in the upstream direction and in the induction field [13].

2.1.2.2 Assumption of incompressibility

Both in momentum and blame element theory the flow is assumed to be incompressible. This assumption is generally considered to have a small effect on accuracy since the maximum Mach number achieved during the operation of a typical turbine is below 0.25. The main reason why the operation of common turbines is restricted at low tip speed rations and thus low Mach numbers is due to noise nuisance. Thus, offshore wind turbines, which are not bound by noise restrictions, are generally able to operate at higher tip speed ratios than common turbines and thus are expected to reach higher Mach numbers. Having said that, even for offshore turbines operating at higher tip speed ratios, the underlying error of the incompressibility assumption is considered to be minor when compared to other sources of error. [13]

2.1.2.3 Representing a rotor as an actuator disc

In BEM theory, an important simplification is made by representing the rotor with an actuator disc. An actuator disc is assumed to be operating in a uniform flow and basically emulates the operation of a rotor with infinite number of blades. In reality, though, a rotor consists of a finite number of blades causing an azimuthally non-uniform flow. The most common correction used in BEM theory to account for this simplification is the Prandtl root/tip loss correction. By employing this correction, a connection is made between the local axial induction of the blades and the azimuthally averaged axial induction factor which is used in momentum theory equations. The validity of Prandtl tip loss correction has been previously investigated through experimental and other more advanced numerical models (AWSM) and some improvements on the original formulas have been proposed to increase the accuracy of the correction. [13]

2.1.2.4 Assumption of independent annuli

As explained in subsection 2.1.1, one of the basic steps of the momentum theory is the breakdown of the stream-tube in several annuli which are assumed to be independent from each other. By making this assumption, it is basically inferred that there is no radial flow present during the operation of the turbine. Such an assumption does not hold when the turbine is operating in yaw since there is a radial component of the velocity present in the rotor. [13] Besides the case of yawed flow, in the work of Sørensen and van Kuik [15], the general momentum theory equations were examined and a large radial pressure gradient has been found especially for rotors operating in low rotational speeds. This existence of a large pressure gradient in the radial direction is in violation with the assumption of annuli independency.

2.1.2.5 Turbulent wake state

The formulation of the equations in subsection 2.1.1 is based on the assumption that the flow inside the stream-tube is going from left to right, i.e. always has a positive direction.

However, this assumption stops to be valid for values of induction factor higher than 0.5. In fact, for $\alpha > 0.5$ the velocity of the flow in the far wake has a negative direction violating the initial assumption. This condition where the flow is reversed is known as turbulent wake state and is often dealt with by incorporating some empirical models in the BEM equations. There have been several empirical models proposed over the years for this issue. Indicatively, some of the most used are the models of Anderson [16] and Glauert [17].

The numerous empirical models available for turbulent wake state indicate that there is a big uncertainty in these corrections. However, since most of the turbines operate close to their optimal tip speed ratio with the goal to achieve the optimal induction factor $\alpha \approx 1/3$, turbulent wake state hardly ever happens and thus there does not seem to be a big need for a correction. In fact, in most codes a correction for turbulent wake state is mostly included to deal with convergence issues and not with the actual phenomenon.

2.1.2.6 Making use of two-dimensional airfoil aerodynamics

As mentioned in previous sections, BEM makes use of $2D$ aerodynamic coefficients to describe the aerodynamic loading of the various blade sections. These coefficients are commonly taken from $2D$ wind tunnel measurements and, thus, are not able to accurately describe the aerodynamic loading of a $3D$ rotating blade that generally has some extra geometric complexity (e.g. taper and twist).

Wind turbine blades are subjected in a specific rotation and it has been observed that this has an effect on the stall behavior of the blades. In fact, rotation seems to be delaying the boundary layer separation at the inner parts of the blade [13]. Several corrections have been proposed over the years to account for this lift/drag increase caused by stall delay. One of the most used and validated corrections is the one of Chaviaropoulos and Hansen [18] who used CFD data to derive a correction coefficient that can be used to generate the 3D C_l and C_d from the respective 2D coefficients.

Besides the effect of rotation on the aerodynamic loading of the blade sections, 3D geometric characteristics like taper and twist are also creating discrepancies from the employed 2D airfoil coefficients. These 3D blade characteristics mostly affect the loading at the tip. In fact, if no corrections are used to account for these 3D effects an overprediction of lift has been observed near the blade tip. A tip correction for such phenomena has been proposed in [19] and has been successfully validated in IEA tasks 14/18 and 20.

2.1.2.7 Assumption of steady flow

One of the most important assumptions made in the derivation of BEM equations is undoubtedly the assumption of stationarity. Both blade element theory and momentum theory are making use of this assumption and thus the introduced error may be quite large if the turbine is operating in unsteady conditions.

From an aerodynamics perspective, the unsteady phenomena present in the operation of a

wind turbine can be divided in unsteady airfoil and unsteady rotor (dynamic inflow) aerodynamics. These two unsteady phenomena have different time scales and thus can be viewed as independent from each other. [13] In fact, the time scale of unsteady rotor aerodynamics is generally much slower.

Unsteady airfoil:
Unsteady phenomena occur at the airfoil level both at stall conditions and when the flow is attached. Both unsteady conditions can be explained in terms of the shed vorticity leaving the airfoil and the change in the angle of attack caused by the shed vorticity. Shed vortices generated by the airfoil are traveling downwind with a specific speed. Thus, when the flow conditions or angle of attack change in time different shed vortices are generated and transported downwind, leading to an airfoil wake constituted by both old and new shed vortices of different strengths. When the old vortices are transported far away from the airfoil, the new vortices dominate the field and the airfoil forces are stabilized.

When the flow is attached, unsteady effects are basically caused by changes in velocity or mild changes in the angle of attack. One of the most known engineering add-ons that have been used over the years to account for unsteady airfoil phenomena at attached conditions is the Theodorsen model [20]. Other more accurate models have also been proposed [21] however they are still not fully validated [13].

When close to stall conditions, dynamic stall unsteady phenomenon is taking place. A sudden change in the angle of attack causes a build-up of vorticity close to the leading edge, leading to a sudden increase in lift. The large vortex is then transported downwind leading to a sudden drop in lift. This increase and decrease of lift is basically creating a hysteresis loop between the lift and the angle of attack. Typically, the dynamic stall models used as BEM add-ons are expressing the lift coefficient as a time-dependent correction of the steady state 2D lift coefficient, based on an ordinary differential equation.

Unsteady rotor aerodynamics - dynamic inflow:
One of the most important assumptions made in the momentum part of BEM theory is that the induction coefficient follows the axial force coefficient instantaneously through Eq. 2.5. However, this assumption breaks down when a sudden change in the axial force of the blades takes place. This sudden change can be caused by several reasons for typical turbines, with the most common being sudden pitch angle, wind speed (gust) or rotational speed changes.

When the loading of the blades changes suddenly, the bound vortices on the blade also experience a sudden change in their strength. The bound vortices corresponding to the new loading are transported in the wake in the form of trailing vortices which are also of different strength than the preexisting trailing vorticity of the wake. Consequently, the wake now consists of both old and new trailing vortices, as shown in Figure 2.5. Since the induced velocity is computed based on the vorticity of the wake, the induction coefficient is not following the new loading situation instantaneously and will only reach the steady-state value of Eq. 2.5 when the “old” vortices travel a considerable distance behind the rotor. The end-

result of this so-called "dynamic inflow" phenomenon is that the induction appears to have a delay to the loading variations on the blade.

The most common way of modeling dynamic inflow in BEM equations is to add a first order time derivative on the induced velocity of Eq. 2.5 and modify the equation as follows:

$$\tau \frac{du_i}{dt} + 4u_i(1 - u_i) = V_w C_{F_{ax}} \quad , \tag{2.16}$$

where τ is smaller closer to the tip blade sections and increases with the rotor diameter. Extensive research has been performed for the validation of such models [13], with IEA Task 20 measurements being the main validation tool. Several results from various research efforts have also been presented in [13] concerning dynamic inflow. Specifically, it has been found that both rotational speed and pitch angle variations can lead to the generation of the dynamic inflow phenomenon, while wind speed variations (gusts) don't really affect the induction field. However, in a latter study by Hammam [22] it has been argued that if the wind speed variations are fast enough, dynamic inflow will occur.

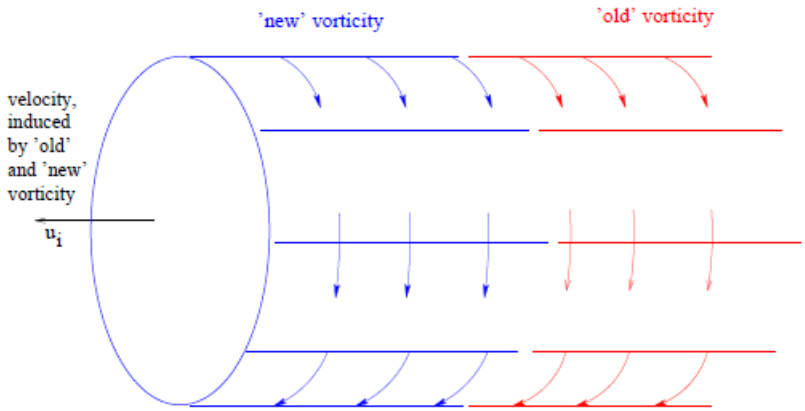


Figure 2.5: Wake consisting of both old and new vortices due to a sudden change in the loading of the blades. [13]

2.2 Computational Fluid Dynamics (CFD)

Computational fluid dynamics (CFD) method aims to predict the flow behavior of a fluid through a numerical simulation. Specifically, their goal is to find an approximate numerical solution to the Navier-Stokes partial differential equations which dictate the physics of the fluid. Generally, the Navier-Stokes equations are broken down to three different partial differential equations which dictate the conservation of mass, momentum and energy, respectively. However, in most wind turbine CFD simulations the flow is assumed to be incompressible and any thermal effects are neglected. Thus, the form of the Navier-Stokes(NS) equations

that are usually solved are as follows:

$$\vec{\nabla} \cdot (\vec{u}) = 0, \tag{2.17}$$

$$\frac{\partial(\vec{u})}{\partial t} + (\vec{u} \cdot \vec{\nabla})\vec{u} = -\frac{1}{\rho}\vec{\nabla}p + \nu\vec{\nabla}^2\vec{u} + \frac{\vec{f}}{\rho}. \tag{2.18}$$

The final solution of these equations are velocity and pressure fields which describe the physics of the fluid flow in 3D space. However, the complexity and non-linearity of these equations mean that the only way to acquire a solution is by discretizing them and then solving numerically. One of the most famous discretization methods that is used by almost all CFD codes is Finite Volume Methods (FVM). FVM basically descritizes the 3D fluid domain in numerous non-overlapping volumes and then solves the conservation equations in their integral form for each volume.

One extra complexity of solving the Navier-Stokes (NS) equations is the existence of turbulence in the flow. There are three methods to account for turbulence when solving the NS equations:

1. Direct Numerical Simulation (DNS): No models are employed for turbulence and the Navier-Stokes equations are solved directly even for the smallest turbulent scale present in the flow.
2. Large Eddy Simulation (LES): The large turbulent eddies of the flow are solved directly while the smaller ones are modeled.
3. Reynolds-Averaged Navier-Stokes (RANS): All turbulent eddies regardless of their scale are modeled and a solution for the mean flow is found.

Due to its low computational cost compared to the other two methods, RANS is currently the industry standard in almost all types of applications including wind turbine simulations. Turbulence models like $\kappa - \epsilon$, $\kappa - \omega$ or others are employed to account for turbulence and a solution for the mean flow is found.

2.3 Important unsteady phenomena for FOWT

The two types of unsteady aerodynamics taking place during the operation of a wind turbine have been already discussed in subsection 2.1.2.7. In this section, vortex ring state and propeller state phenomena are discussed. These two unsteady aerodynamic phenomena are of particular importance for the operation of floating offshore wind turbines.

A wind turbine can operate in various induction factors. Depending on the value of the induction factor the nature of the turbine's wake is changing, as shown in Figure 2.6. For $\alpha < 0$, the thrust force on the rotor is negative and the rotor is operating in propeller state accelerating the flow that passes through [23]. For $0 < \alpha < 0.5$, the rotor is operating in regular windmill state and all the assumptions of momentum theory are valid. For $0.5 < \alpha < 1$, the wake enters the so-called turbulent wake state and the assumptions of the momentum theory are not valid anymore [24]. Last but no least, for $\alpha = 1$, the streamtube enters vortex ring state where the flow passing through the rotor is blocked and huge recirculation occurs normal to the disc.

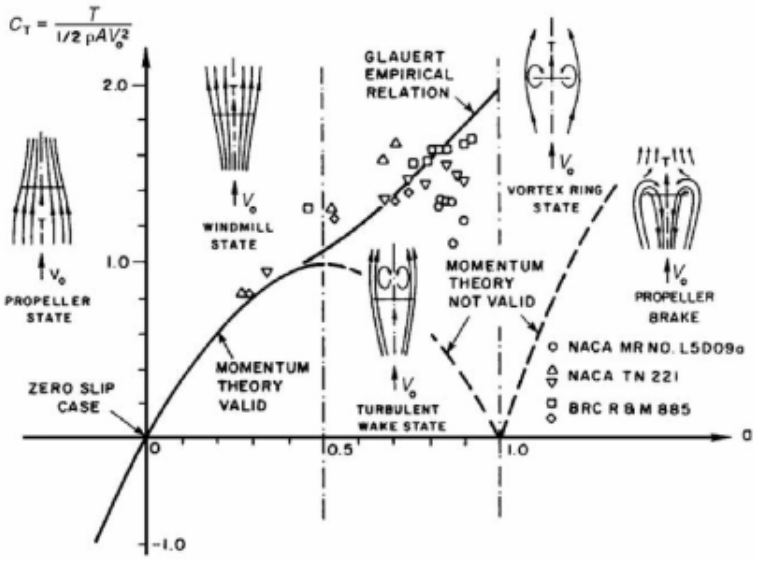


Figure 2.6: Streamtube wake states depending on the induction factor. [25]

Typical wind turbines are not expected to experience propeller or vortex ring states because they are fixed on the ground. On the contrary, as discussed in section 1.3.2, floating offshore wind turbines experience several motions during their operation and thus may experience such phenomena. According to Sebastian and Lackner [8], a FOWT under pitch motion will interact with its wake during its motion, and thus, vortex ring and propeller state can occur, as shown in Figure 2.7. As the turbine moves in the downstream direction it enters its own wake. If the rotor is moving in the downstream direction with a large velocity, the effective wind speed experienced by the rotor may drop significantly, causing toroidal recirculation which is usually linked with vortex ring state. In the work of Kyle et al [26] the same topic is investigated by performing a CFD simulation for a rotor under intense surge motion. When noticing negative thrust coefficient values Kyle et al concluded that propeller state occurs during the motion of the rotor. Furthermore, they argued that when the thrust acquires negative values tip and root vortices are accumulated and can not be advected in the wake leading to a state that based on the authors prove the existence of vortex ring state during surge.

If aerodynamic phenomena like vortex ring state and propeller state actually occur during the

operation of FOWT, then all the industrial aerodynamic codes that employ momentum theory basically break down. Thus, there is a big concern in the wind energy community regarding the applicability of the current industrial engineering codes for FOWT. Having said that, there is still a big confusion in the scientific community on how exactly these wake states are identified. In the work of Dong and Vire [27], various criteria that could identify vortex ring state are proposed. Moreover, in [28] Ferreira et al argue that negative rotor thrust does not necessarily mean that the streamtube of the rotor enters propeller state. In fact, they continue by saying that the flow in the streamtube does not have the time to adjust to the negative loading of the rotor and remains in wind-mill state throughout the surge motion.

In the present thesis, a harsh wave state is also investigated. Induction factor and thrust calculation is carried out throughout the surge motion period. Furthermore, since blade-resolved CFD data are produced, the complex 3D flow field can also be visualized to answer the critical question of whether propeller or vortex ring state really occur.

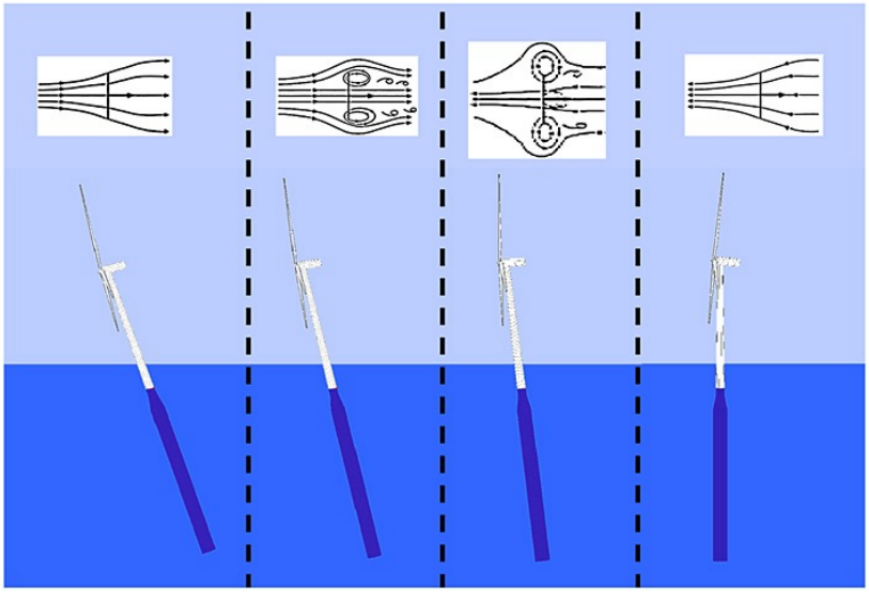


Figure 2.7: Wind turbine under pitch motion and the hypothesized wake states occuring during its motion. [8]

2.4 BEM for the aerodynamic analysis of FOWT

The low computational cost and the simplicity of BEM make it up to this day the most widely used method to assess the aerodynamic performance of wind turbines. However, as discussed in the previous subsection, several assumptions are made in the derivation of this simple theory and, thus, several uncertainties are introduced in the solution. For each of these assumptions, engineering add-ons have been included in the theory to remedy the introduced error and make it possible to use BEM for the complex designs of modern wind turbines.

In the case of floating offshore wind turbines, extra motions and moments are added into system, as shown in section 1.3.2. The absence of a rigid pylon let the turbine to be subjected into these motions, with surge and pitch being the ones exhibiting the largest displacements. The displacement of the rotor during its normal operation is increasing the aerodynamic unsteadiness, due to dynamic inflow phenomena, and may affect its aerodynamic performance. Thus, a question arises whether BEM and its present dynamic inflow engineering add-ons are adequate for the analysis of FOWT. Significant research has been performed the last 10 years to answer this question, with most of the authors employing higher fidelity codes or experimental measurements on FOWT under surge.

The first attempts to investigate the unsteadiness imposed by surge motion on floating wind turbines was done using the benchmark case of a 5MW NREL turbine. Several authors investigated the same baseline model and even though they studied the same case, their conclusions were somewhat contradictory. Sebastian and Lackner [8] employed a free vortex wake code and were the first to point out that the common BEM unsteady corrections for dynamic inflow are not sufficient to accurately model the unsteadiness of a FOWT. On the contrary, de Vaal [29] compared the results of a moving actuator disc model to a quasi-steady BEM model and a simple BEM model with an Øye [30] correction and concluded that the typical frequencies of a TLP floater are not high enough to create dynamic inflow effects. Following the work of de Vaal, Micalef and Sant [31] investigated the same frequency range to de Vaal and found deviations between the results of BEM, a generalized dynamic wake and an actuator disc code. In the same work, unsteady dynamic inflow effects were observed to increase in strength with an increasing tip speed ratio. This observation is quite important considering that offshore turbines are typically operated at higher tip speed ratios than common turbines. The same observation was also validated in the work of Farugia [32] who analyzed the wake of a FOWT under surge motion using a free vortex wake code. Lastly, considering the same NREL 5MW FOWT case, Tran and Kim [33] employed a high-fidelity RANS CFD simulation to study the same frequency range with the aforementioned authors. Their work concluded that for high surge frequencies, the results of BEM deviate from the ones of the CFD model.

Another recent big research effort on the unsteady aerodynamics of FOWT under surge motion was the UNAFLOW project [34]. A scaled version of a DTU 10MW turbine was examined and the low-fidelity BEM code results were compared to high-fidelity numerical codes like CFD and actuator line (AL) models. However, the major difference between UNAFLOW and the previous research efforts was that the results of the various numerical models were also compared to high-fidelity wind tunnel measurements focused entirely on the aerodynamics of FOWT. The conclusion of the UNAFLOW project was that all numerical codes were in good agreement with each other and also with the experimental measurements. In fact, even the BEM code with a traditional dynamic inflow model was able to capture the loads of the rotor and produce results of acceptable accuracy. The unsteady phenomena did not appear to have a significant effect on the loads, which is also in line with the results of de Vaal [29] who supported that a quasi-steady or a simple BEM model (with an Øye correction) is able to capture the loads of a FOWT under surge motion. Having said that, researchers of UNAFLOW point out that for higher frequencies than the ones they examined more significant unsteady effects may appear. In fact, examining a higher frequency range is one of the future goals of these researchers.

The contradictory conclusions of the aforementioned research efforts on the accuracy of BEM exhibits that there is still much research to be done regarding the unsteady aerodynamics of FOWT. High-fidelity codes and wind tunnel measurements is of paramount importance in order to validate the codes of lower fidelity like BEM. However, the end-goal of all the research efforts is to find an accurate engineering model with low computational cost that will be able to capture the complex aerodynamics of FOWT.

2.5 CFD for the aerodynamic analysis of FOWT

Computational fluid dynamics (CFD) was always imperative for any type of aerodynamic research since it's the most accurate numerical model available to the scientific community. In the case of floating wind turbines and the investigation of their aerodynamic performance, CFD becomes even more important. Together with high-fidelity experiments, CFD is the only way to acquire high-fidelity data regarding the complex aerodynamic phenomena taking place during its operation. Significant research efforts have been done on this subject over the last 10 years with the goal to fully explain the physics of this complex system. In the present section, the latest CFD research efforts related to FOWT are presented. After reviewing the literature related to CFD on FOWT, the goals and methodology of the present master thesis will be investigated.

2.5.1 Importance of CFD

An accurate cost estimate analysis for modern wind farms is not a simple process. First and foremost, structural models with high accuracy are needed to predict the fatigue loads experienced by the turbines. In turn, structural models take as input the aerodynamic forces computed from the aerodynamic models. Thus, high accuracy aerodynamic models are not only important for computing the energy output of a turbine and the nature of its wake but also for the computation of the fatigue loads. Furthermore, since wind farms consist of several wind turbines in a specific layout, controllers that regulate the rotational speed, yaw angle and pitch are crucial for an efficient energy production with the least possible fatigue loads on the turbines. Last but not least, meteorological models are needed to predict the wind in specific sites which is basically the first input in the cost estimate analysis.

Taking into consideration the numerous models that have to be included in a cost estimate analysis, it is easy to understand why industry still prefers low-fidelity BEM aerodynamic models instead of high-fidelity CFD codes for the prediction of WT aerodynamics. As explained in section 2.1, BEM has a low computational cost and can be easily coupled with the various other codes needed in a cost estimate analysis. Having said that, CFD is indispensable not only for the validation of BEM engineering add-ons but also for giving us a better insight of the physics of the various complex aerodynamic phenomena. In fact, in order to produce these engineering add-ons a detailed insight of the physics is needed beforehand.

In the case of FOWT, the later argument becomes even more important since the floater motions introduce an extra level of complexity and uncertainty in the unsteady aerodynamics

of the turbine. Even though there have been several research efforts on this subject the last few years, the physics of such a complex system are not fully understood yet. In fact, the contradictory conclusions discussed in subsection 2.4 indicate that there is still much research needed on the subject and thus the importance of high-fidelity aerodynamic data from CFD simulations and experimental measurements become even more important to fully apprehend the physics and reach to solid conclusions.

2.5.2 CFD research efforts for FOWT

Significant research has been done over the last 10 years in the field of floating offshore wind turbines. Nevertheless, various questions still remain unanswered regarding the operation of FOWT, with the researchers mostly focusing on the structure, aerodynamics or controller related issues.

In the case of FOWT aerodynamics, both numerical and experimental studies have been performed but the majority of publications are employing low-fidelity numerical codes like BEM, FVW, and GDW to analyze the aerodynamic performance. Due to the increased complexity of FOWT, only a few researchers are making use of CFD to acquire high-fidelity results while even fewer researchers are employing high-fidelity experiments [35]. As discussed in section 1.3.2, the floating platforms of FOWT are mostly excited by surge and pitch motions. Thus, most CFD publications are investigating the aerodynamic performance of FOWT under surge, pitch or a combination of both movements.

Tran et al [33] employed a URANS overset grid CFD simulation and a conventional BEM code to analyze the aerodynamics of the NREL 5-MW baseline WT model under surge motion. Significant deviations were found between the low-fidelity BEM results and the CFD results, with a percentile difference on the power coefficient reaching even 30% for the highest amplitude and frequency of the sinusoidal surge motion applied to the system. Furthermore, it was observed that the turbine wake was significantly affected by the motion of the rotor and the tower, causing a deviation of its strength which increased with an increasing frequency/amplitude of the surge motion.

Under the framework of the UNAFLOW research project, Cormier [36] and Mancini [37] employed a state-of-the-art URANS CFD overset grid simulation which is probably the most extensive work for the investigation of FOWT unsteady aerodynamics. The unsteady aerodynamics of a scaled version of the DTU 10MW turbine to a harmonic surge motion was examined in great detail. First, in the work of Cormier [36], a one-third axisymmetrical model was investigated consisting of one blade and one third of the hub. Together with CFD, FVW and BEM codes were employed to represent a lower-fidelity numerical result, while high-fidelity experimental measurements were used to validate all codes. In all CFD studies in the literature, the chosen frequency and amplitude of the harmonic surge motion applied to the rotor are the major factors determining the unsteadiness of the floating system. Thus, in the UNAFLOW project, a non-dimensionalised indicator of unsteadiness, f_{red} , called the reduced surge frequency, which was first introduced by Bayati et al [36], was also employed to characterize the unsteadiness of the imposed conditions on the rotor. The definition of the unsteadiness indicator can be expressed as follows:

$$f_{red} = \frac{f_s D}{V} \quad , \tag{2.19}$$

where V is the inflow wind velocity, f_s is the frequency of the surge motion applied to the floating wind turbine and D is the diameter of the rotor. This metric basically compares the frequency of surge to the characteristic one of dynamic inflow, which is the most relevant source of unsteadiness associated with floater motions. The higher this metric is the larger the effects of dynamic inflow phenomena on the response of the turbine. Specifically, for $f_{red} < 0.2$, the aerodynamics can be considered as quasi-steady whilst the higher the value of f_{red} the more non-linear unsteady aerodynamic effects take place. The one-third model without tower and nacelle of Cormier [36] served as the initial validating stage for the CFD model, nevertheless its results showed a remarkable agreement with both the low-fidelity codes and the high-fidelity experiments, despite the fact that highly unsteady conditions of $f_{red} = 0.59$ and $f_{red} = 1.19$ have been chosen. Continuing the work of Cormier and investigating exactly the same operating conditions, Mancini et al [37] built a more accurate and complete CFD model that incorporated the tower and the nacelle in the geometry but also used a higher-order spatial discretization scheme and accounted for wall blockage that should be present in the wind tunnel measurements. This work exhibited once more a very good agreement among all numerical codes and experimental measurements and showed that the unsteadiness of the rotor did not affect the loads significantly. In fact, the mean values of the unsteady loads were in good agreement with the equivalent steady values and, thus, even the conventional BEM code with a simple dynamic inflow model was able to capture the loads accurately. Having said that, as pointed out by Mancini et al, higher values of the reduced frequency that will lead to further unsteadiness on the rotor may lead to significantly different results.

URANS CFD simulations were also employed by researchers to investigate other important motions besides surge. In the work of Chen [38], the NREL 5 MW baseline turbine was subjected to surge, pitch or a combination of both motions. Similarly, in the work of Wu [39], URANS CFD was employed to examine the performance of NREL 5MW turbine under pitch, surge or yaw. The works of both authors only analyzed the rotor-hub configuration without the tower and the nacelle and concluded that, similarly to surge, pitch motion also leads to large oscillations of the rotor loads. Chen et al [38] focused on the effect of a combined surge-pitch motion on the performance of the rotor and one of the main conclusions was that the power generation of a turbine under the combined motion is up to 6.32% smaller than the turbine that is just under pitch motion. In fact, the majority of the authors in the literature report that the power coefficient seems to be more sensitive to surge motion than the thrust coefficient. Besides pitch and surge, yaw motion was also investigated by Wu [39] but it did not seem to have a significant effect on the performance of the rotor.

Despite their higher computational cost, CFD simulations can provide information about the flow that simply can't be computed by conventional lower-fidelity numerical codes like BEM. Kyle et al [26] employed a URANS CFD simulation to investigate whether the full configuration of the NREL 5MW turbine can experience propeller state and vortex ring state (VRS) under surge motion. Their work showed that the turbine experienced these states under some favorable surge frequencies and amplitudes. In fact, the operating conditions

used in the work of Kyle lead to a reduced frequency of $f_{red} = 2.22$ which translates to a highly unsteady state for the rotor. Another remarkable work was that of Fang et al [40] who employed an improved delayed detached eddy simulation (IDDES) to examine the performance of a 1:50 model FOWT under surge and the wake recovery process of such a turbine. IDDES is practically a combination of RANS near the wall regions and LES in the far field regions. By employing this hybrid approach, the authors investigated the field around a rotor-hub configuration and were able to accurately compute the far wake region with LES while also computing the field close to the walls at a low cost. Two important conclusions were drawn from this analysis. First, for the highest surge frequencies and amplitudes examined, the rotor exhibited an increased average rotor power up to the order of 40% when compared to the fixed rotor power. Second, the wake recovery of the rotor under surge seemed to be slower than the rotor without surge.

An important point to be made about the work of Wu [39], Chen [38], Fang [40] and Kyle [26] is that due to the lack of experimental FOWT measurements the CFD models are validated by lower-fidelity numerical codes like FVW and BEM. In fact, most of the CFD works in the literature are validated with lower-fidelity data since the access to high-fidelity experimental data is very limited. One of the few works that included high-fidelity experiments in their analysis is the work conducted by the UNAFLOW project. Thus, their work will serve as the basis of the present thesis and the validation of the CFD model will be performed based on their high-fidelity experimental data.

2.6 Intermediate aerodynamic models between BEM and CFD

The high computational cost of CFD simulations and the low accuracy of blade element momentum theory led the scientific community to intermediate solutions, i.e. to numerical codes that serve as the middle ground by acquiring results of higher accuracy than BEM and lower computational cost than CFD simulations. Currently the most famous alternatives to BEM and CFD are free vortex wake (FVW), actuator disc (AD) and actuator line (AL) codes. In the present section, these three intermediate codes will be briefly discussed and the main research efforts related to FOWT under surge will be investigated.

2.6.1 Free wake vortex (FVW)

Free wake vortex codes are based on the assumption of potential inviscid flows and cylindrical wake. In FVW codes, the blades are treated as lifting lines with a specific bound vorticity computed from lift through Kutta-Joikowski theorem. Similarly, the wake is represented by trailed and tip vortex lines that are transported downstream of the blade lifting lines. Similarly to BEM, in order to acquire the lift that is initially needed for the definition of the blade's bound vorticity, airfoil 2D polar data are needed. The induction field is eventually computed from the vortex lines of the wake by making use of Biot-Savart law.

Even though this model is a substantial upgrade from BEM in terms of accuracy for unsteady cases, it still has several limitations. The assumption of inviscid flow, the inability to model inflow turbulence and the employment of 2D steady airfoil data to model 3D blade aerodynamics make the performance of this intermediate numerical code questionable especially when wind farm performance prediction is needed.

In the literature related to FOWT under surge motion, Sebastian [41] employed a FVW code and was one of the first authors to indicate that the accuracy of BEM codes may not be adequate to capture the aerodynamics of FOWT. Farrugia et al [32] also employed a FVW code to investigate FOWT under surge and came to the conclusion that the unsteadiness of the flow increase with an increasing tip speed ratio of the rotor. Although FVW delivers a better accuracy level than BEM, in most research projects it is mostly incorporated to represent the low-fidelity results together with BEM. Specifically, in the UNAFLOW project FVW was found to have relatively good agreement with other numerical codes and experimental data when it came to the unsteady thrust response. On the other hand, the power generation predicted by FVW for the steady non-surging turbine was off by a factor of 10% when compared to high-fidelity experimental data.

2.6.2 Actuator disc (AD) and actuator line (AL) models

Both actuator disc (AD) and actuator line (AL) models are solving the Navier-Stokes equations just like in the case of CFD simulations. However, in contrast to pure CFD, AD and AL are modeling the blades of the turbine with a disc or a line, respectively. By modeling the aerodynamic forces on this disc/line, the need to resolve the boundary layer of the blades is eliminated and, thus, the computational cost is reduced substantially. At the same time since most of the computational cost is avoided, higher accuracy scale-resolving simulations like LES can be performed instead of RANS. In fact, most AD and AL simulations are paired with an LES solver to investigate the turbine wake in great detail. Having said that, the fundamental lift and drag forces of the blades are taken from 2D airfoil polar data just like in BEM. Thus, even though these methods deliver good accuracy in wake reproduction, they usually make an error in the computation of the integral loads and especially in thrust.

Being methods of intermediate accuracy, AD and AL are usually employed in the literature to validate the less accurate BEM method. In the works of de Vaal [29] and Micallef [31], a moving actuator disc model was employed to validate whether BEM can capture the aerodynamic of a FOWT under typical surge frequencies/amplitudes using a common dynamic inflow model or a quasi-steady assumption. Furthermore, as in the work of Farrugia [32], Micallef [31] also found an increase of unsteadiness with an increasing tip speed ratio. In contrast to AD, actuator line (AL) codes have not been used that much for the aerodynamic analysis of FOWT under surge. In fact, the work of Mancini both in [37] and out [35] of the UNAFLOW project is one of the few works where AL was included in the analysis. In this work, AL appeared to be in good agreement with higher-fidelity CFD and experimental data.

2.7 Surge conditions examined in the literature

In this section, the main results of the literature are gathered. The purpose of gathering all results is to show the various surge parameters that were investigated by the authors and reveal which surge conditions are less examined and leave room for further investigation.

2.7.1 Representation of surge motion in the literature

In all the studies of FOWT under surge motion found in the literature, the surge motion is represented by a sinusoidal signal. Furthermore, in almost all studies the phase of the sinusoidal signal is set as 0 to further simplify the motion of the rotor. In the present thesis, the same sinusoidal signal will be employed and can be written as follows:

$$x_s = -A_s \ \sin(\omega_s t) \quad , \tag{2.20}$$

$$V_s = \frac{dx}{dt} = -\omega_s \ A_s \ \cos(\omega_s t) \quad , \tag{2.21}$$

where x_s is the surge motion displacement, V_s the surge motion velocity, A_s the surge motion amplitude, and ω_s the surge motion frequency in rad/s. The surge motion displacement and velocity have a pure sine and cosine shape as shown in Figure 2.8.

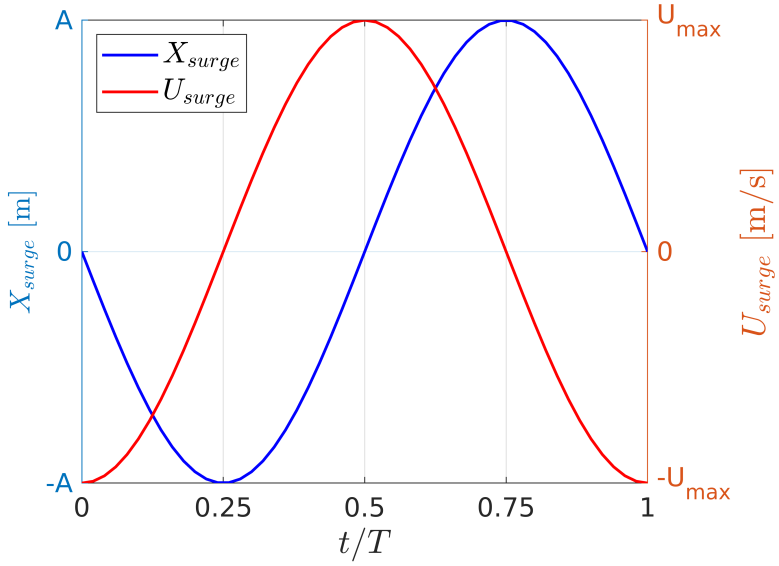


Figure 2.8: Surge motion displacement and velocity represented by sinusoidal signals.Graph taken from [28].

2.7.2 Reduced surge motion variables

As discussed previously, several surge motion frequencies and amplitudes have been investigated in the literature. To be able to compare the work performed by different studies, it is important to define reduced frequencies and amplitudes of the surge motion. The reduced frequency f_{red} has already been introduced in the UNAFLOW project [20] and has been discussed in section 2.5.2. In the work of Ferreira et al [28], similar reduced variables have been introduced and will also be employed in the present thesis to choose the operating conditions of the CFD simulations. Specifically, the reduced frequency ω_{red} and the reduced amplitude A_{red} of the surge motion are defined as follows:

$$\omega_{red} = \frac{\omega_s D}{V} \quad , \tag{2.22}$$

$$A_{red} = \frac{A_s}{D} \quad , \tag{2.23}$$

where ω_s is the frequency of the surge motion in rad/s, D is the diameter of the rotor , A_s is the amplitude of the surge motion and V is the free-stream velocity. Combining these two reduced variables, the reduced maximum velocity of the surge motion can be acquired:

$$V_{max,red} = A_{red}\ \omega_{red} = \frac{A_s\ \omega_s}{V} \quad . \tag{2.24}$$

Reduced maximum velocity shows practically how the maximum velocity of the surging rotor compares to the incoming freestream velocity and is a very important variable because it represents the relative velocity of the rotor at an extreme position during a surge period. For a surging rotor, the axial velocity V_{axial} right before the rotor can be described as follows [29]:

$$V_{axial} = V - V_s - V_{in} \quad , \tag{2.25}$$

where V_s is the velocity of the surging rotor, V_{in} is the induced velocity and V is the free-stream velocity. When the rotor is under surge, the velocity of the rotor oscillates, as it was shown in Figure 2.8. Since the surge velocity oscillates, V_{axial} also oscillates leading to an oscillation of the loading. In fact, in Figure 2.9 Ferreira et al [28] gathered the results of the literature in a simple graph which is showing that a more adverse velocity oscillation, i.e. a larger V_{max}, also leads to a larger oscillations of thrust in the rotor. Therefore, it is very important to compare the results of the various studies while taking into account the respective $V_{max,red}$ that they examined.

In Figure 2.10, the reduced frequencies and amplitudes examined by various authors in the literature are presented. Reduced maximum surge velocity $V_{max,red}$ is also shown at the

same graph in the form of isolines. It can be seen that most of the work done in the literature concentrates on mild wave-state conditions that lead to a reduced maximum velocity of $V_{max,red} \leq 0.1$. In contrast, surge conditions that lead to a $V_{max,red} \approx 1$ are much less investigated. Examining the aerodynamics of a surging rotor with $V_{max,red} \approx 1$ is very interesting because in such extreme conditions the relative velocity experienced by the rotor becomes 0 or even lower than 0 leading to negative thrust values. As discussed in section 2.3, there is ambiguity in the wind energy community on whether negative thrust values indicate that the wind turbine enters into propeller state. One of the goals of the present study is to contribute in clearing the ambiguity on this topic.

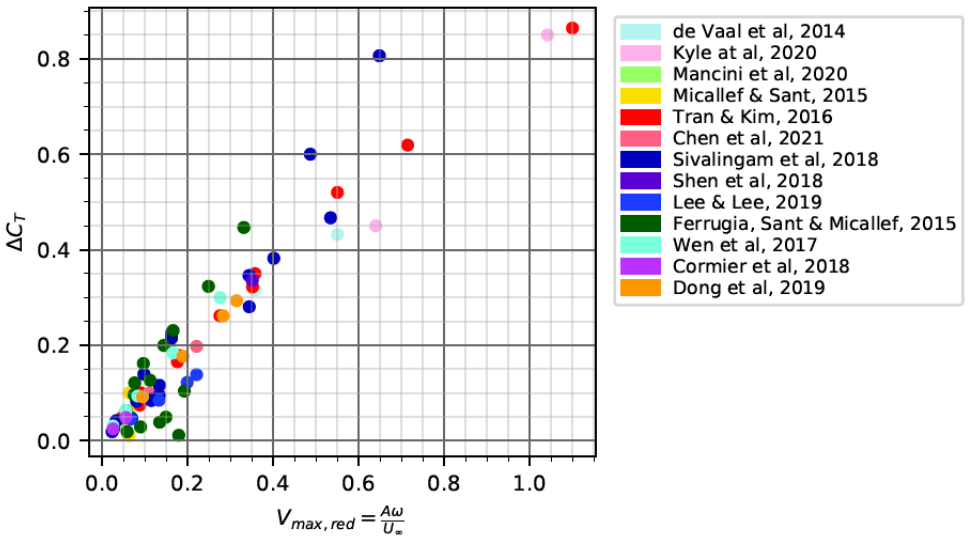


Figure 2.9: Response of the amplitude of thrust coefficient to $V_{max,red}$. Graph taken from [28].

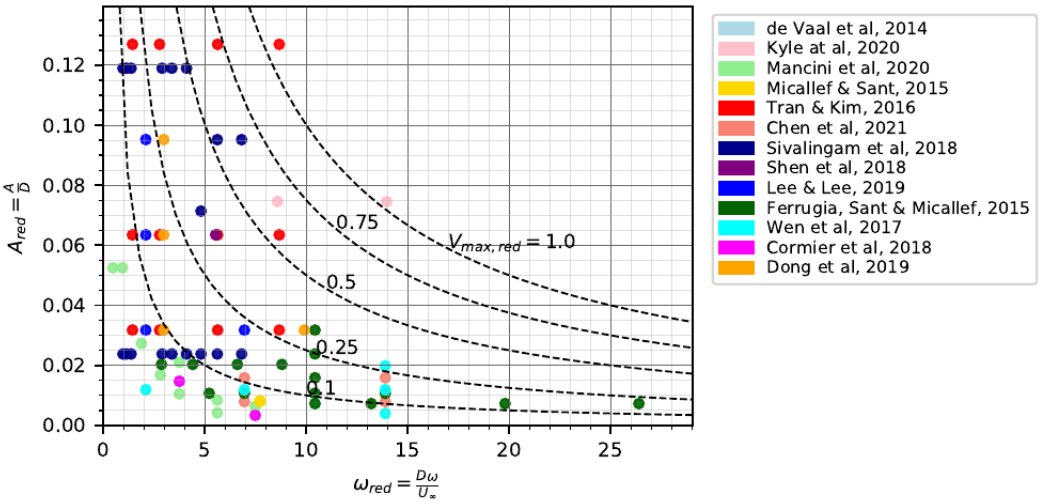


Figure 2.10: Surge amplitudes and frequencies examined in the literature. Reduced maximum velocity $V_{max,red}$ is also shown in the form of isolines. Graph taken from [28].

Chapter 3

Thesis Project Plan

In the present chapter, the plan of the thesis is organized. There are several decisions that have to be made prior to building the CFD model. First, the wind turbine type (e.g. NREL 5MW, DTU 10MW etc.) and the specific wind turbine setup (e.g. rotor-alone,rotor+nacelle+tower etc.) are chosen. Next, the operating conditions and the surge motion frequency/amplitude are chosen. Last, the general outline of the simulations that will be carried out is presented using a workflow chart.

3.1 Choosing the wind turbine that will be investigated

One of the most important steps when performing a CFD analysis is the validation of the computational model. Through the validation of the CFD model we build trust on the produced data and make sure its accuracy will be adequate enough to answer the research questions in the best possible way. As discussed in section 2.4, the experimental data of the UNAFLOW project is, to our best knowledge, the only high-fidelity experimental measurements for FOWT under surge. Thus, in order to be able to validate the results of the CFD model with high-fidelity data, the turbine model used in the UNAFLOW project will also be employed for the computations of the present thesis. Specifically, a 1:75 scaled version of the DTU 10 MW turbine is chosen which was designed by Polimi to match exactly the thrust coefficient of the DTU turbine. The basic characteristics of both the real turbine and its scaled version are shown in Table 3.1. The exact geometry characteristics of the blades, tower, nacelle and hub of the Polimi model can be found in the work of Bayati et al [42].

An important decision that has to be made at the very start of the CFD model setup is the type of geometry that will be simulated. In the literature associated with CFD simulations on FOWT under surge, some authors take the full turbine configuration with tower and nacelle while others prefer to simplify the problem and simulate only the rotor with the hub. In the present thesis, the rotor and hub configuration will be chosen for the sake of lowering the

Table 3.1: Basic characteristics of the DTU 10 MW turbine and its scaled 1:75 model used in the UNAFLOW project.[37]

	DTU 10 MW RWT	Polimi model
Control	Variable speed + collective pitch	Variable speed + individual pitch
Drivetrain	Medium speed, multiple-stage gearbox	Transmission belt, epicyclic gearbox
Gearbox ratio	50	42
Diameter	178.3 m	2.38 m
Hub height	119 m	2.05 m
Tilt angle	5°	5°
Coning angle	−2.5°	0
Blade prebend	3.33 m	0

already high complexity of the problem. The very first step of the present thesis is to compare the CFD results with the high-fidelity experiments of UNAFLOW. If large discrepancies are found between our CFD results and the UNAFLOW experimental data, the full turbine configuration with tower and nacelle might be taken to improve the accuracy of the CFD model.

3.2 Test cases

Besides choosing the same turbine model, the same operating conditions with the ones examined in the UNAFLOW project will be the first to be tested in order to validate the CFD model. In fact, the CFD model has to be validated both in a fixed turbine setting without the surge motion and in a turbine setting under surge motion. The full test matrix that was followed in the UNAFLOW project can be found in the thesis work of Mancini [35]. From the 91 sets of operating conditions tested in the UNAFLOW project only two of them were also simulated with their CFD model. One of these two testing conditions, namely $f_s = 2Hz$ & $A_s = 0.008m$, will be tested in the present thesis so that the CFD model of the thesis will also be compared to the results of the UNAFLOW CFD model.

Apart from the operating conditions tested for validating purposes, a higher surge frequency and surge amplitude, resulting to a $V_{max,red}$ slightly higher than 1, will be examined to represent a rougher sea state that could lead to higher unsteadiness of the system. As discussed in section 2.7.2, there are not many authors in the literature that investigated surge conditions leading to $V_{max,red} \approx 1$. Such conditions could lead to propeller or vortex ring states and, thus, it is of significant importance to investigate whether such conditions really occur since they practically mean the breakdown of typical aerodynamic models used in the industry. The final test matrix of the present thesis is presented in Table 3.2. In this table, the first two sets of operating conditions are performed to validate the CFD model while the last set is performed to answer the research questions of the thesis.

Table 3.2: Test matrix of the present thesis.

#	CFD Model Type	V_0 [m/s]	λ [-]	Ω [rpm]	θ_p[deg]	f_s [Hz]	A_s [m]	$V_{max,red}$ [-]
0	Steady-State	4	7.55	242.4	0	0	0	0
1	Unsteady	4	7.55	242.4	0	2	0.008	0.025
2	Unsteady	4	7.55	242.4	0	5.5	0.120	1.036

3.3 Outline of the CFD simulations that will be performed

After selecting the wind turbine setup and the operating conditions that will be tested, the outline of the simulations that will be carried out in the present thesis can now be presented in the form of a workflow chart, as shown in Figure 3.1.

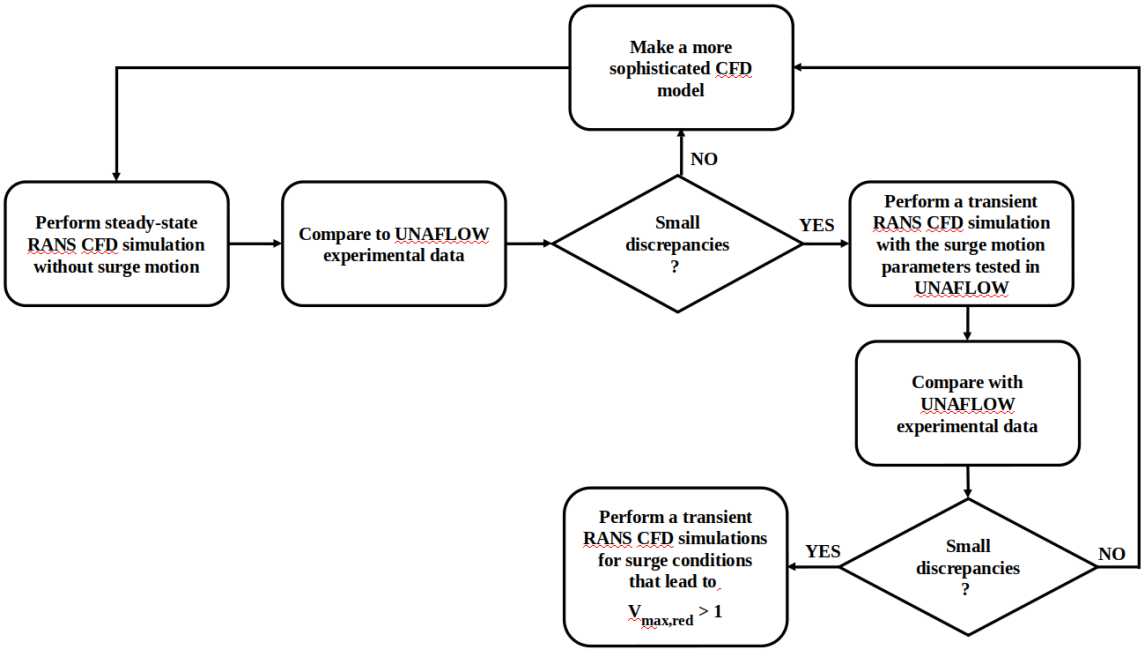


Figure 3.1: Outline of the CFD simulations that will be performed

Chapter 4

Methodology

In this chapter, the methodology of the present thesis is presented. First, the decisions that have to be made to close the CFD model are presented. Next, the meshing process is described and a mesh-independence study is performed to make sure that the simulation results are not dependent to the mesh resolution. Last but cerntainly not least, a validation process is followed where the CFD model is compared to the UNAFLOW experiments both for a surging and a non-surging rotor.

4.1 Simulation Set-Up

For both the steady-state and the unsteady simulations that are going to be performed, some fundamental parameters have to be chosen to close the CFD model and make sure that the model will be accurate and efficient enough to produce the desired data. Each and every decision that has to be made regarding the CFD model is discussed based on the literature and an argument is made on which is the most appropriate choice for the present thesis.

4.1.1 CFD software

Before moving into the details of the CFD model, a CFD software has to be chosen for the present thesis. Several CFD software have been used by researchers in the literature to study the unsteady aerodynamics of an FOWT. Tran [33] and Fang [40] used Siemens software STAR-CCM+, Chen [38] used ANSYS FLUENT, Kyle [26] utilized the open-source software OpenFOAM, while for the UNAFLOW project an in-house CFD solver called FLOWer was employed. For the present project, OpenFOAM will be used because it is an open-source solver that can be modified according to the needs of the project and no license is needed to perform the simulations. These two characteristics make OpenFoam a good choice for a thesis project since the CFD model produced by the thesis can be easily passed on and

then modified by other students for the purpose of their thesis. Specifically, the version of OpenFOAM released by ESI-OpenCFD will be chosen due to its slightly better documentation and slightly better ability to simulate multiple moving bodies.

4.1.2 3D Domain & Boundary Conditions

In this subsection, a description of the computational domain as well as the boundary conditions applied on its boundaries are given.

As discussed previously, the rotor that will be simulated in the present thesis is the one employed in the UNAFLOW project. In this project, an experimental campaign was performed in the boundary layer test section of the Polimi's wind tunnel GVPM. Specifically, the test section was 13.84 m wide, 35 m long and 3.84 m high. The scaled wind turbine of UNAFLOW was positioned at the center of test section's height and the wind speed experienced by the rotor was constant with a very low turbulent intensity (TI) of about 2%.

In order to validate the CFD model of the present thesis with the results of the UNAFLOW experimental campaign, the computational domain of the simulations will follow the geometry of the wind tunnel test section, as shown in Figure 4.1. The only difference with the geometry of the GVPM test section is that the computational domain length is a bit shorter at 17.8m. The distance of the wind turbine rotor from the inlet and the outlet of the domain was chosen to be 6 rotor radii and 9 rotor radii, respectively, according to the work of Sayed [43] who investigated the influence of boundaries' distance to the rotor's forces.

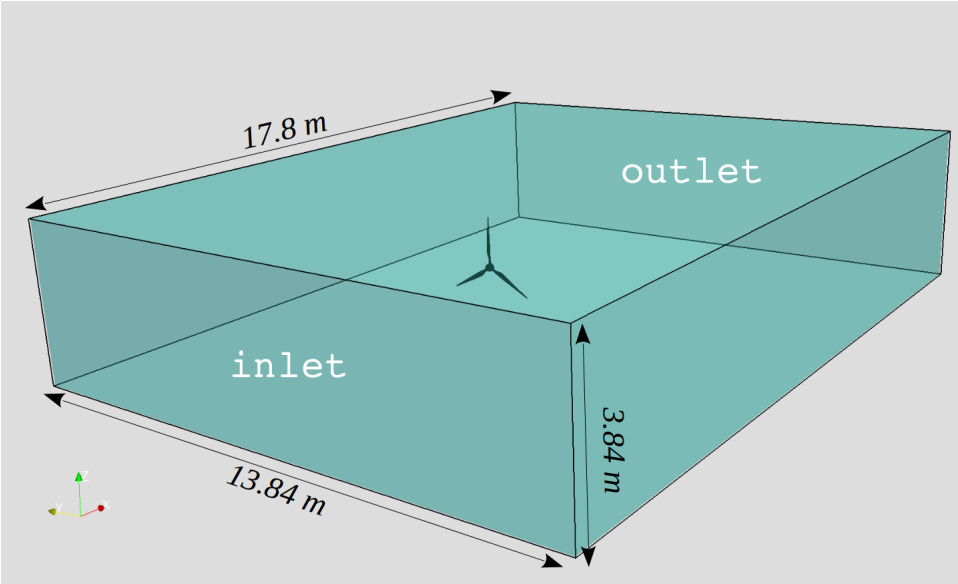


Figure 4.1: 3D Domain of the CFD model, rotor positioned 6R from the inlet.

Besides the inlet and the outlet of the computational domain, all the other side boundaries shown in Figure 4.1 are treated as wall boundaries since GVPM test section is a closed test

section. The boundary conditions imposed on the boundaries of the computational domain are:

1. Inlet: Dirichlet velocity boundary condition $\vec{u} = (4, 0, 0)$ and Neumann pressure boundary condition $\frac{\partial p}{\partial n} = 0$
2. Outlet: Dirichlet pressure boundary condition $p = 0$ and Neumann velocity boundary condition $\frac{\partial u}{\partial n} = 0$
3. Sides: slip walls boundary condition, i.e. $\vec{u} \cdot \vec{n}$ and $\frac{\partial u}{\partial n} = 0$

4.1.3 Type of CFD simulation

Due to the surge motion of the floating platform and the resulting motion to the rotor of the wind turbine, a highly unsteady flow field is created. Such a flow field can not be captured by a steady- state solver and can only be fully resolved with a transient solver. Thus, a transient solver will be employed for the present thesis. As discussed in section 3.2, unsteady simulations will be performed with two surge motion frequencies/amplitudes, while a steady-state solver will only be used at the validating stage of the CFD model where the rotor will be simulated without a surge motion.

Furthermore, a Reynolds Averaged Navier Stokes (RANS) solver will be preferred against a more computationally expensive LES solver since the complexity of the problem is already quite high. Unsteady Reynold Averaged Navier Stokes (URANS) solvers were also employed by almost all authors in the literature, with only Fang [40] using a hybrid of LES and RANS, namely IDDES.

4.1.4 Turbulence modelling

Another very important choice that has to made to close the computational model is the turbulence model. Different turbulent models may lead to different results depending on the type of flow field of the simulation, thus extra caution has to be taken when choosing the turbulence that will be used. Most of the times the turbulence model of CFD simulations is chosen based on prior experience on how a specific model performed in solving a similar problem. Thus, the best way to decide which turbulent model to choose is to search the literature for the choices that other researchers made and the end-result they had in their simulation. When searching the literature related to FOWT under surge, $k-\omega$ SST turbulence model stands out as the most frequent choice of the researchers. In fact, all the works mentioned in section 2.5.2 employ this turbulence model. The $k-\omega$ SST turbulence model is practically a combination of $k-\omega$ and $k-\epsilon$ models, and its goal is to take advantage of the benefits each model has to offer [44]. In the near-wall region, the $k-\omega$ model is used while in the far-field and outer boundary layer regions the $k-\epsilon$ model is used. By combining these two models, the inaccuracy of the $k-\epsilon$ in the near-wall region as well as the sensitivity of $k-\omega$ to inlet turbulence boundary conditions are neutralized and only the best of the two worlds are kept.

Besides choosing the turbulence model that will be used, the turbulence boundary conditions need to be assigned both for inlet/outlet faces and for sides/blades/hub walls. Specifically, the turbulence boundary conditions applied to the boundaries are as follows:

1. Inlet:
   (a) k: "fixedValue" condition with $k = 9.6 \times 10^{-3}$.
   (b) ω : "fixedValue" condition with $\omega = 328.76$.
   (c) ν_t : "calculated" condition.
2. Outlet:
   (a) k: "InletOutlet" condition.
   (b) ω : "InletOutlet" condition.
   (c) ν_t : "calculated" condition.
3. Sides:
   (a) k: "slip" condition.
   (b) ω : "slip" condition.
   (c) ν_t : "slip" condition.
4. blades & hub:
   (a) k: "fixedValue" condition with $k = 1 \times 10^{-12}$.
   (b) ω : "omegaWallFunction" condition with $\omega = 3.2 \times 10^{8}$.
   (c) ν_t : "nutLowReWallFunction" condition.

All the conditions are specific OpenFoam conditions and their function can be found in [45]. The values of $k = 9.6 \times 10^{-3}$ and $\omega = 328.76$ at the inlet are approximated with the following formulas:

$$k = \frac{3}{2}(I\,|\mathbf{u}|)^2 \quad , \tag{4.1}$$

$$\omega = \frac{k}{\nu}\left(\frac{\nu_t}{\nu}\right)^{-1} \quad , \tag{4.2}$$

where I is the turbulent intensity, $|\mathbf{u}|$ is the mean flow velocity and $\frac{\nu_t}{\nu}$ is the eddy viscocity ratio. Turbulent intensity is of the order of 2% following the experiment of UNAFLOW, while the eddy viscosity ratio is set as 2 which is supported to be a good choice for the inlet of external aerodynamics simulations [46].

By the turbulence boundary conditions applied for the blade and hub walls, it is apparent that a wall-resolved simulation will be performed, aiming for a $y^+ \leq 1$. Carrying a wall-resolved

simulation with a $y^+ \leq 1$ increases the computational cost substantially due to an increase in the number of cells close to the walls. Having said that, a wall-resolved simulation is the only way to accurately compute the forces in the present project, since the surge motion imposed on the rotor may lead to separation in several regions of the blade.

4.1.5 Spatial and temporal discretization schemes

Another choice that has to be made to close the CFD model is the type of discretization scheme that will be used for resolving the convective and the diffusive terms of RANS but also the scheme for time marching. A scheme that is readily available in OpenFOAMv2012 and is of second-order accuracy is the second-order upwind scheme. This scheme has been proven to be an efficient solving technique of the incompressible RANS equations when moving bodies are involved in the simulation [47, 48]. In fact, most of the authors in the literature related to CFD for FOWT have taken the second order upwind scheme for their models, with the exception of Mancini [37] who made use of a more accurate but at the same time more computationally expensive fifth-order weighted non-oscillatory (WENO) scheme. Since a second-order scheme was chosen for the spatial discretization, it is always a good practice to keep the same level of accuracy in all levels of the CFD model, thus a second-order temporal discretization should be chosen. From the options readily available in OpenFOAMv2012, the Crack-Nicolson second-order bounded implicit scheme seems like the best choice for solving the incompressible URANS equations.

Even though second-order accuracy was the goal of the present thesis, numerical stability issues for the unsteady simulation led to a blending between first-order and second-order for the time discretization scheme and increased non-orthogonality corrections for the laplacian term. Specifically, after testing 20 different configurations for the schemes, second-order upwind was kept as it was for the divergence term, CrankNicolson with a blending factor of 0.5 was chosen for the time scheme and a Gauss Linear scheme with a blending factor of 0.33 was chosen for the laplacian term.

Another choice that has to be made is the value of the time-step that will be employed. At the very first unsteady simulations that were performed the time-step was chosen based on the value of the CFL number. Using the PIMPLE algorithm of OpenFoam and aiming for a CFL≤ 2 led to a time-step of $dt = 10^{-7}$. Such a low time-step translates to an excruciatingly slow simulation that would need about 1 year of computational time to produce the desired results. Upon further investigation it was found that CFL was close to 2 only for the first boundary layer cell located exactly at the sharp corner of the trailing edge while the remainder of the mesh had a CFL value much smaller than 1. This observation indicated that a larger time-step could be chosen since CFL would be larger than 2 only for the first cell at the sharp T.E. corner. Furthermore, several researchers in the literature have based their simulation time-step choice based on the rotor rotation degrees per time-step and not based on the CFL number. In fact, a value smaller than $\theta \leq 1^\circ$ of rotor rotation per time-step was found by several researchers in the literature [36, 37, 38, 39, 26, 40] to be small enough to capture the unsteady aerodynamics of FOWT under surge. After taking into account the aforementioned arguments, the final value of the time-step that was chosen for the present thesis was $dt = 9 \times 10^{-6}$ which translates to 0.013° of rotor rotation per time-step and leads

to a computational time of about 7-9 days. A much larger time-step could be chosen and still capture the physics of the problem, however, numerical stability issues limited the time-step to this value.

4.1.6 Resolution of the moving bodies

The biggest complexity of a simulation for a FOWT under surge motion is the fact that there are multiple motions that the solver should account for. Besides the rotating blades, the floater imposes a translational oscillating motion to the whole configuration. Therefore, the rotor experiences one rotational and one translational motion at the same time. The most common ways to resolve moving bodies in CFD are summarized in this section and an argument is made on which is the best choice for the present project.

- **Multiple Reference Frame (MRF)** The MRF approach is a fairly simple modeling technique that is most commonly used to solve rotating machinery problems. This technique is usually employed to get the steady-state loads of a turbine, propeller etc. without the need to run an unsteady simulation and account for moving bodies with a moving mesh of some type. When the MRF technique is applied to account for the motion of a rotating rotor, a volumetric zone is created around the rotor which is also called the MRF zone. Outside the MRF zone there is a common stationary zone where there is no rotation. In this technique, no actual mesh movement is utilized for the MRF zone. On the contrary, the steady Navier-Stokes equations are manipulated and a source term is incorporated on the right- hand side of the equation which practically accounts for the rotation of the MRF zone. This source term only exists for the MRF zone while when moving in the stationary zone of the computational domain the source term vanishes. Considering the logic behind this technique it is easy to understand that it can only be used when the time averaged steady-state solution of the flow around a single rotating rotor is needed.

- **Sliding Mesh Technique** The sliding mesh technique allows two adjacent mesh zones to move (slide) relative to each other. The two mesh zones have a common interface on top of which one or both of the mesh zones are moving. Since there is a sliding motion of two mesh zones on top of this interface the two mesh zones are obviously not conformal and an interpolation technique is needed to transfer information and preserve continuity between the two mesh zones. The interpolation is done at the mesh interface. In OpenFOAM, this patch type is called arbitrary mesh interface or AMI. This technique is employed when the transient solution is needed in a simulation that involves a relative motion between two regions. A common example of where the sliding mesh technique could apply can be shown in Figure 4.2. In fact, sliding mesh technique could be applied in any type of simulation where there is a rotation or translation between two separate mesh regions and the topology of the two regions will not change during the motion.

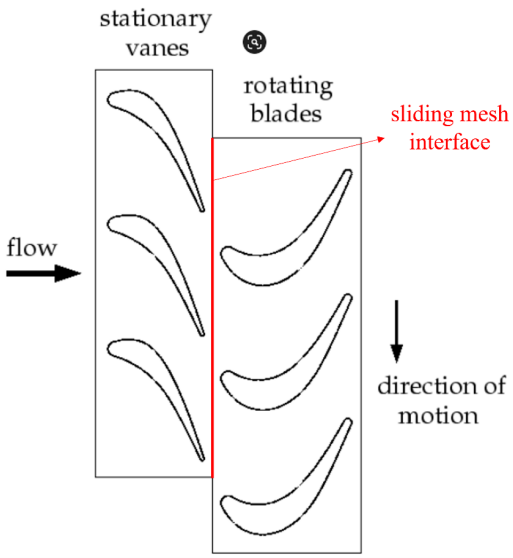


Figure 4.2: Simulation of a rotor-stator configuration employing a sliding mesh technique.[49]

- **Morphing Mesh technique** Another way to resolve moving bodies is by making use of deformation and morphing techniques. There are several techniques of this type but all of them work in the same way. Specifically, assuming that there is a prescribed motion in the simulation, at each timestep the body has to be transported from an initial to a final position in the computational domain. When adopting a mesh deformation technique, the mesh cells are following the body to its new position by a suitable deformation and morphing of the mesh which is controlled by an algorithm incorporated in the solver. This technique is generally considered to be computationally expensive since the mesh has to be deformed at each timestep of the simulation. Having said that, there are several occasions where this technique will be able to produce much more accurate results than SRF, MRF or sliding mesh technique.

- **Mesh Motion Technique** In this technique, we aim at moving the whole domain. In contrast to morphing mesh and sliding mesh technique a rigid solid body motion is applied to the cells of the whole domain and no topology changes occur in the cells. This kind of technique is particularly useful when dealing with sloshing tanks cases where a motion is applied to the whole tank.

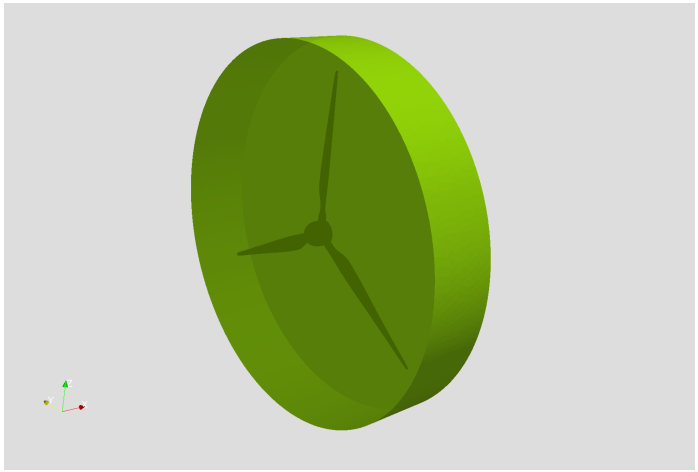
- **Overset Mesh Technique** Overset meshes which are also known as "Chimera" meshes are employed when complex mutli- geometry motions are present in a simulation. The grid zones around the moving bodies are built separately and superimposed on the background stationary mesh. Several different meshes can overlap each other in a random way, thus overset meshes can pretty much capture any type of arbitrary motion of one or multiple bodies. In fact, overset meshes are ideal for simulations with complicated body motions that involve bodies that are moving in a close distance with each other and possibly have intersecting paths. Each simulation that involves overset meshes has a background mesh region and one or more smaller mesh regions that overlap the background mesh. The smaller mesh regions are free to move in any type of desired motion and thus during their motion the overlapping regions between the background and the

moving meshes change. In an overset mesh the computational cells are mapped into active, inactive and acceptor cells. Inside active cells the governing equations are solved while in inactive cells the equations are not solved. Inactive cells are basically the cells that lie inside the solid body and the cells mapped as inactive can be changed based on the motion of this body into the background mesh. Lastly, the cells that are mapped as acceptor cells are basically the cells lying on the boundary between the smaller moving regions and the background mesh. Acceptor cells are responsible for interpolating the solution between the background and the smaller overset moving meshes.
The procedure followed by overset mesh technique is performed at each time-step of the simulation and is quite time-consuming. Since this technique substantially increases the computational cost of a simulation, it is used only in cases where there are multiple bodies in motion and there is no other to resolve them.

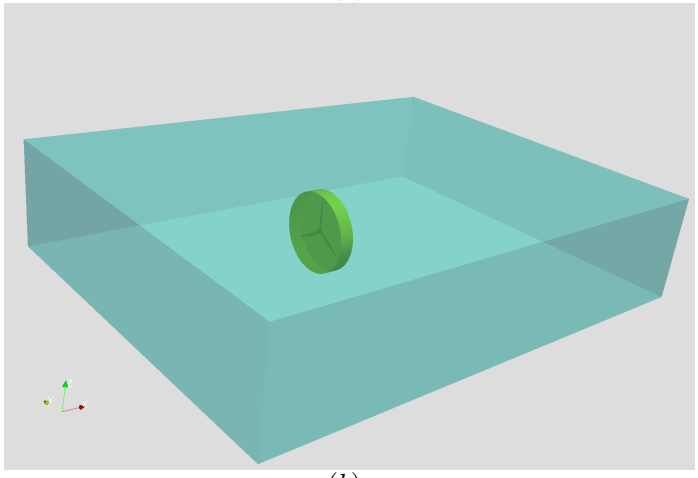
As discussed previously, there will be two kinds of simulations in the present thesis. First, a steady-state simulation will be performed for validation purposes, where its results will be compared with the UNAFLOW experimental results for the fixed turbine. After validating the CFD model, an unsteady RANS simulation will be performed which will also include the surge motion of the turbine. In both types of simulations, the tower and nacelle are omitted to reduce the computational complexity and only the rotor-hub configuration is simulated.

For the steady-state simulation of the rotor-hub configuration, there is only the rotational motion of the blades that need to be resolved. Thus, an MRF technique bundled with the simpleFoam steady-state solver will be employed aiming for a reduction of complexity and computational cost. In this simulation, a rotation is applied on the reference frame of the cells inside a cylindrical zone (MRF zone) enclosing the rotor, as shown in Figure 4.3. Outside this MRF zone the reference frame of the cells experience no rotation. The MRF technique was obviously only used for the steady-state simulation and even though it is expected to be less accurate than any mesh motion technique, it is the fastest way to validate the accuracy of the numerical schemes and the mesh by comparing with UNAFLOW fixed turbine results. Furthermore, by performing a fast steady-state simulation, the initial solution of the transient simulations was acquired.

For the unsteady simulation of the rotor-hub configuration, the rotational motion of the blades was resolved with a sliding mesh technique. The sliding mesh technique is the industry standard for turbomachinery applications due to its simplicity, low computational cost and high accuracy. As with the case of the MRF technique, a cylindrical zone is created around the rotor, as shown in Figure 4.3. The only difference with the MRF is that this time an actual mesh motion is applied for the cells inside the cylindrical zone. Besides the rotational motion, there is also the oscillating surge translational motion that needs to be taken into account. To resolve this motion, the mesh motion technique is applied to the whole computational domain. By applying a motion to the whole domain, we practically impose a translational motion to the global reference frame. Furthermore, since all the cells of the mesh are moving with the exact same velocity, no deformation occurs at the cells and, thus, this procedure is not computationally expensive.



(a)



(b)

Figure 4.3: Cylindrical zone enclosing the rotor. (a): Zoomed in view (b): Zoomed out view

4.2 Constructing The Mesh

In this subsection, the meshing stage of the project is described. Meshing is one of the most time-consuming and complex procedures that need to be followed to construct a CFD model. In the present thesis, the complexity is further increased due to the complex 3D geometry of the blades. In fact, a wind turbine blade has several demanding regions in terms of meshing. A smooth transition of the mesh should be accomplished at the trailing edge, tip and hub junction regions of the blade. Otherwise, highly non-orthogonal and skewed cells will be created at these regions, causing a poor quality mesh and, thus, low quality results or even a divergence of the solver. Besides these critical mesh regions, the boundary layer and the wake regions need to have an adequate resolution to accurately resolve the flow. In the present

thesis, since the accurate computation of the rotor forces is the main objective, the boundary layer resolution is probably the most critical issue that needs to be settled. For this reason, 30 inflation layers are applied on the walls of the blades and the hub while aiming for a first layer height of $y^+ \leq 1$. In order to satisfy the y^+ condition the first layer height needs to be about $y = 5 \times 10^{-6}$ meters. Thus, it is obvious that the resolution of the mesh will be quite large close to the blades and a coarsening of the mesh in the span-wise direction needs to be applied aiming for an aspect ratio lower of about 1000 for the cells of the first layer.

Since OpenFoam was the software chosen for the present thesis, snappyHexMesh, a meshing algorithm bundled with OpenFoam, was the first meshing tool that was used. SnappyHexMesh is an automatic 3D mesher containing hexahedra and split-hexahedra. After creating a background mesh, snappyHexMesh tries to conform the mesh to the triangulated CAD surface following an iterative procedure. Even though snappyHexMesh requires little input from the user and is generally easy to operate after investing about 1-2 weeks to understand the control parameters, it was proven to not be the appropriate tool for meshing the complicated geometry of the blades. Indicatively, some critical regions meshed with snappyHexMesh are shown in Figure 4.4. In all the critical regions, and especially at the T.E. and tip regions, snappyHexMesh failed to built inflation layers and produced some low quality transition regions, mainly due to the desired small first cell height of $y = 5 \times 10^{-6}$ m and a total of 30 layers extruding from the blades' airfoils. After thoroughly researching regarding the capabilities of snappyHexMesh, it was found that meshes aiming for $y^+ \leq 1$ are very difficult to be constructed with this specific tool and even if an experienced user succeeds, the final mesh will have a very high non-orthogonality leading to a mesh with poor quality.

After failing to produce a high-quality mesh with snappyHexMesh, other automatic open-source software were tried out. Salome and cfMesh were the two standing out with the most potential. However, after investing a few weeks with these meshing tools, they also seemed to fail in producing a high-quality mesh with $y^+ \leq 1$.

Since all the efforts to mesh the model with automatic open-source tools were unsuccessful, the commercial software Pointwise was given a try. Pointwise is considered as one of the best meshing software in the CFD community and is also one of the most preferred meshing tools for wind turbine applications. In order to make sure that this time a high quality mesh with $y^+ \leq 1$ would be constructed close to the blades, a multi-block structured mesh was applied for the regions close to the blade. A hybrid approach was then followed, with a tetrahedral unstructured mesh connecting to the structured hexahedral mesh and extending to the far-field of the domain. Even though the construction of the multi-block hexahedral part of the mesh was quite demanding with a lot of manual input, the end-result was a mesh of very high quality and very smooth transitions in the critical mesh regions.

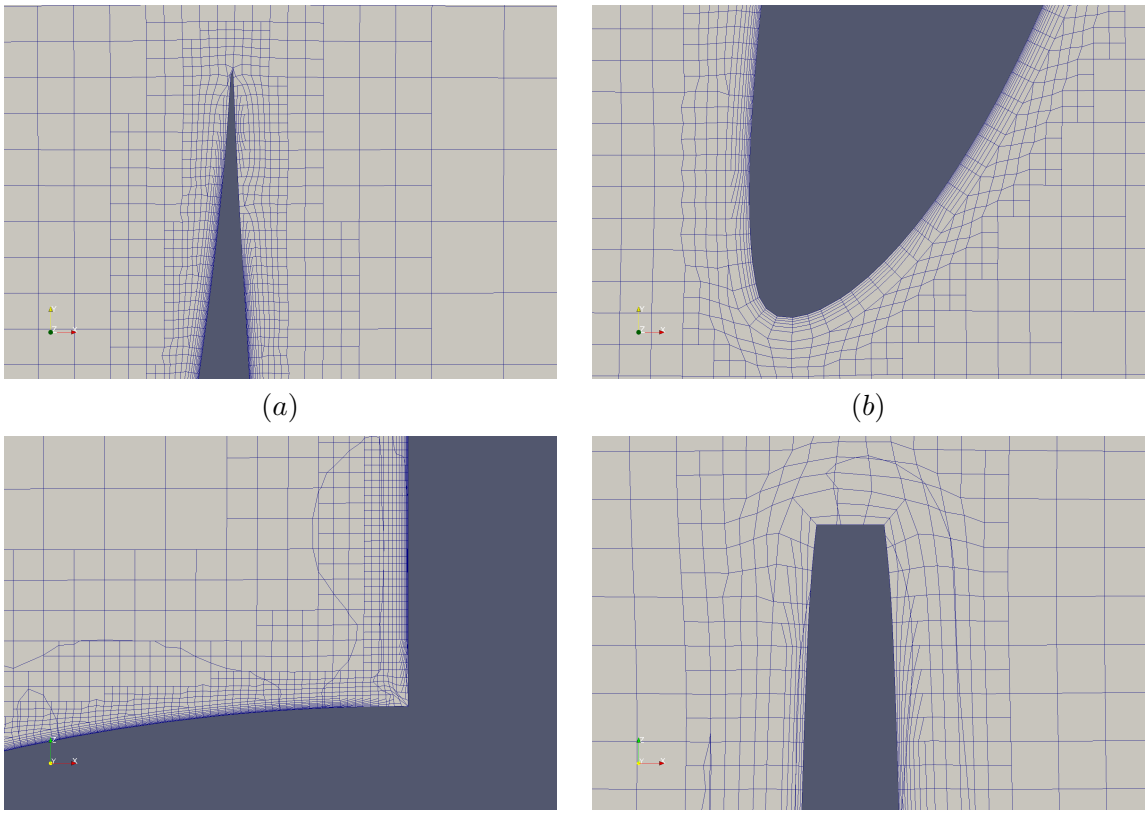


Figure 4.4: SnappyHexMesh meshing efforts. Critical mesh regions close to the blade are shown. (a): T.E region (b): L.E region (c): Blade-Hub junction region (d): Tip region

In Figure 4.5, some general views of the mesh produced with Pointwise are shown. In Figure 4.5(c), it is observed that the tetrahedral mesh starts to refine upstream of the rotor and as it gets closer to the rotor the refinement level increases gradually. In the same view, it can be seen that expanding refinement regions are formed on the wake of the hub and the blade tips. The blade tip refinement region is also visible in Figure 4.5(b) where it is shown that the region creates a cylindrical tube enclosing the tips and making sure that the tip vortices will be resolved as they convect away from the tips.

In Figure 4.7, some critical regions of the mesh are shown. It is observed that in contrast with snappyHexMesh, Pointwise manages to create reliably 30 inflation layers with a first layer thickness of $y = 5 \times 10^{-6}$ in all challenging regions of the blade. Furthermore, it is observed that there is a smooth transition between the cells and that their volume ratio is small. Due to these smooth cell transitions in all regions of the mesh, the grid quality is very high and all the important quality metrics are within optimal ranges. Indicatively, the non-orthogonality metric is shown in Figure 4.6. Non-orthogonality is quite an important metric because it affects the accuracy of the results and the stability of the solver. Non-orthogonality below 75° is considered quite good when solving with OpenFoam while a value above 80° is typically causing numerical instability and divergence of the solver. Thus, with a maximum value of 72° and the vast majority of the cells having a value below 65°, the present mesh

exhibits a very high quality.

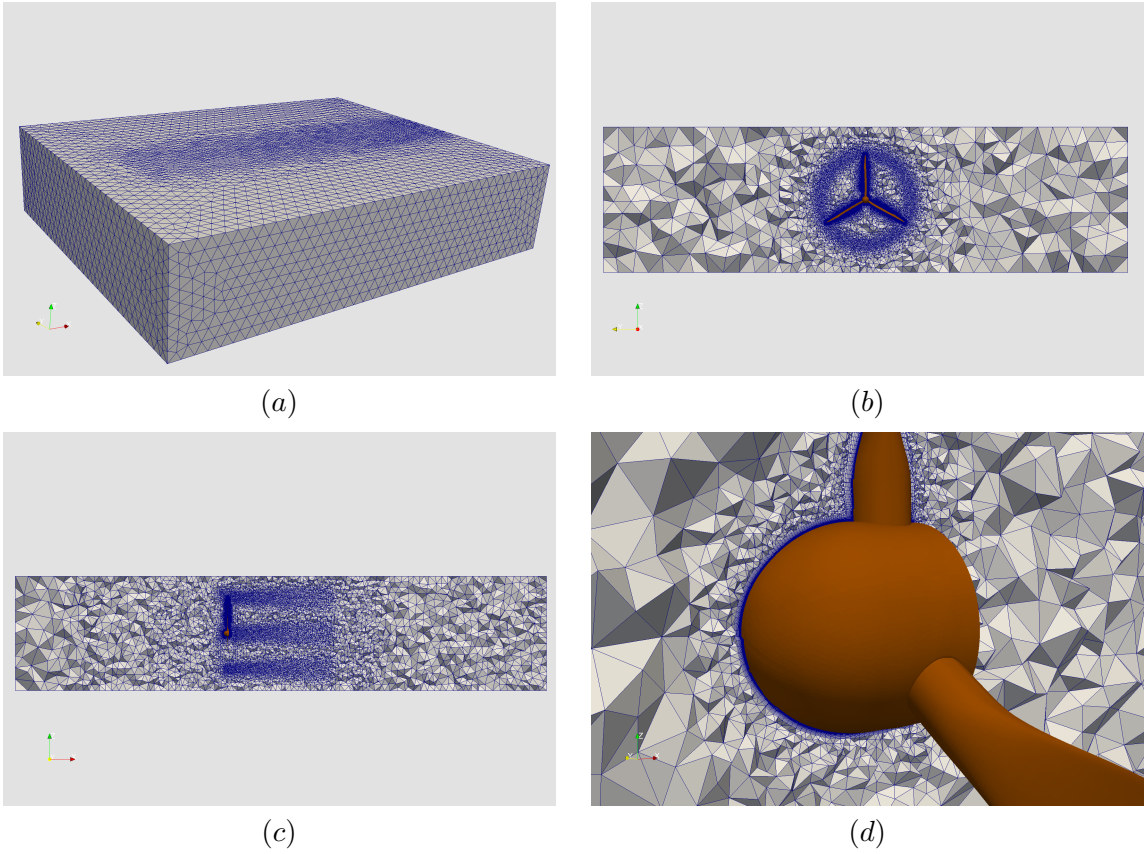


Figure 4.5: General views of the final mesh produced with Pointwise (a): outside the domain (b): $x = 0$ cross-section (c): $y = 0$ cross-section (d): zoomed in view of the blade-hub junction

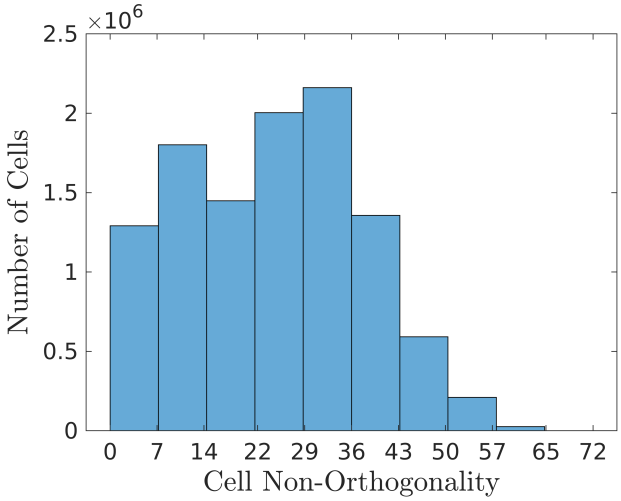


Figure 4.6: Cell non-orthogonality histogram.

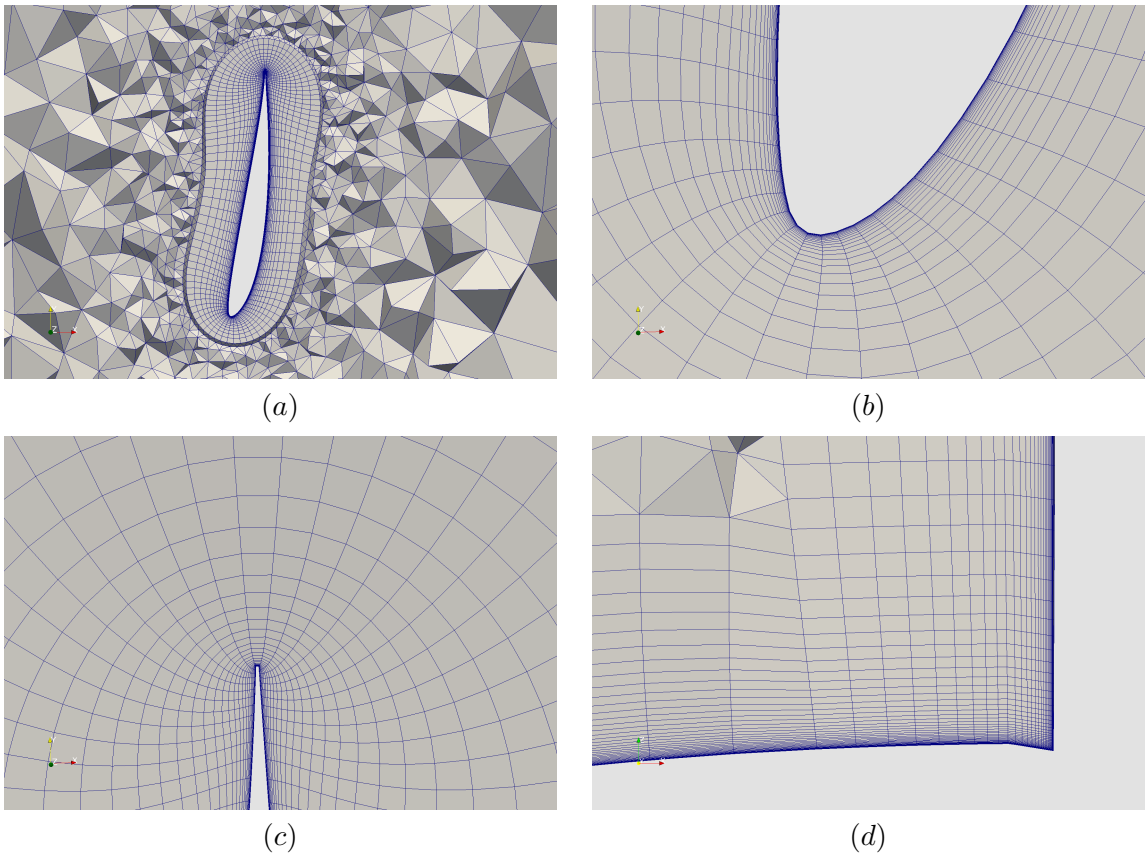


Figure 4.7: Final mesh produced with Pointwise. Critical mesh regions close to the blade are shown. (a): blade cross-section (b): L.E region (c): T.E region (d): Blade-hub junction region.

4.3 Computational resources

After constructing the mesh and setting up the simulation parameters the CFD model is closed and simulations can now be initiated. As discussed previously, both steady-state and transient simulations were performed in the present thesis. Steady-state simulations are known to be much less computationally intensive than transient simulations, however a decomposition of the domain is several processor regions was needed for both types of simulations.

For the steady-state simulations, where no mesh motion is involved and the solver is not advancing in time, the computational domain was decomposed in 20 processor regions with the hierarchical method of the decomposePar command of OpenFoam. After decomposing the domain, a job was submitted on the TU Delft cluster hpc12 and the simulation was started in one Intel(R) Xeon(R) CPU E5-2670v2 node which belongs to the group of fpt-small nodes. Running in 20 logical processors of the E5-2670v2, the steady-state simulation converged after about 3 days (72 hours).

For the transient simulations, mesh motion and time-stepping are enabled which means that

20 logical processors are not sufficient any more for such a computationally intensive job. After testing several decomposition methods of OpenFoam and running for logical processors ranging from 96 to 480, the most efficient choice of resources was chosen. Specifically, the hierarchical decomposition method proved itself to be very inefficient and slow for such a large number of processsor regions, and thus, the domain was decomposed in 448 processor regions with the scotch method of OpenFoam. After decomposing, the simulation was started in 14 Intel(R) Xeon(R) Gold 6130 nodes (32 cores per node) which belong to the fpt-large group of nodes of the hpc12 cluster. Even though such a high processor count was chosen, the transient simulation needed about 14 days (336 hours) to finish. This long running time can be attributed to the very small time-step that was needed to keep the solver stable and the fact that we need the simulation to reach a specific final time of about 2-3 surge periods in order for the solution to reach its final periodic state.

4.4 Grid-Independence Study

After creating a mesh with high quality, a grid-independence study needs to be conducted. In CFD simulations, grid-independence studies are essential to make sure that the results coming out of the simulation are trustworthy and not dependent on the mesh resolution. To perform such a study several meshes with different resolutions are typically constructed and the desired simulation is performed multiple times. The goal of this study is to find the mesh resolution above which results (i.e. thrust, torque, etc.) stop changing and then use it for further simulations.

In the present thesis, the steady-state simulation without the surge motion will be employed for the grid-independence study. In contrast with the transient simulation, the steady-state simulation of the present thesis can yield results in a few hours. Thus, it is preferred over the slow transient simulation, since several simulations need to be performed to find the optimal mesh resolution. Three different meshes are constructed, one coarse with 8.5 million cells, one fine with 20 million cells and one with medium resolution and 11 million cells, as shown in Figure 4.8. The MRF steady-state simulation is performed for all three meshes and the resulting thrust and power of the rotor is computed.

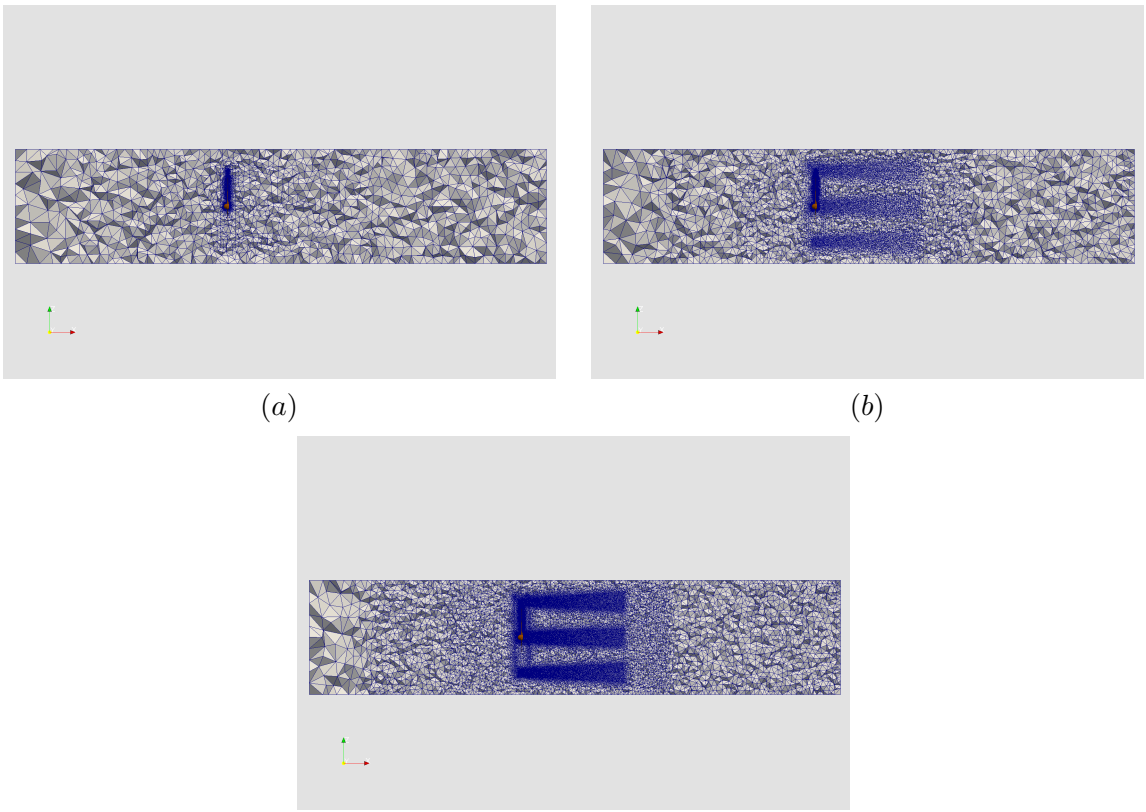


Figure 4.8: Cross-sections of meshes produced for the grid-independence study. Mesh resolution: (a) 8.5M, (b) 11M and (c) 20M.

In Figure 4.9 and Figure 4.10, the total thrust and power of the rotor are presented, respectively. It is observed that, for all three mesh resolutions, the simulations converge after about 1000 iterations to approximately the same value of thrust and power. The percentile differences of thrust/power between the three meshes are presented in Table 4.1. The results of the three meshes are almost identical with percentile differences lower than 0.25%.

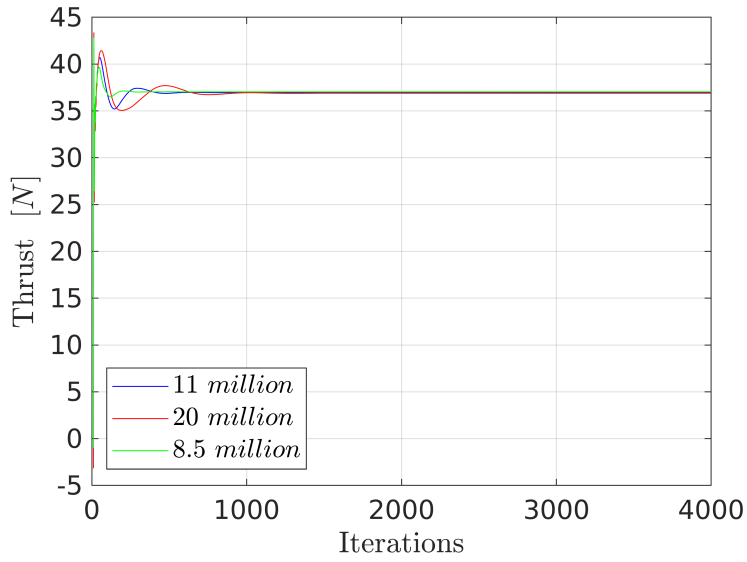


Figure 4.9: Grid-independence study. Thrust force computed for meshes of various resolutions with an MRF steady-state simulation.

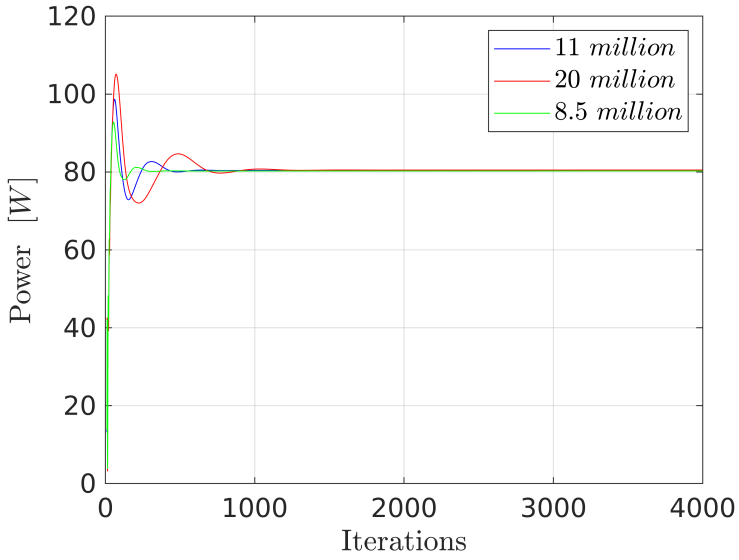


Figure 4.10: Grid-independence study. Power computed for meshes of various resolutions with an MRF steady-state simulation.

Table 4.1: Grid-independence study with an MRF steady-state simulation. Percentile difference of thrust/power between coarse (8M), medium (11M) and fine (20M) meshes.

	8.5M	**11M**	**20M**
T (N)	37.03	36.95	36.89
% difference T	—	−0.21 %	−0.16 %
P (W)	80.27	80.41	80.50
% difference P	—	+0.17 %	+0.11 %

Besides computing the thrust and power produced by the turbine blades, it is also essential to visualize the flow field and see how the wake is resolved for each mesh resolution. In Figure 4.11, the velocity magnitude, V_{mag}, is visualized in a $y = 0$ cross-section of the domain. It is observed that there is excessive smearing of the solution for the coarsest mesh, while for the medium and fine meshes the flow field is almost identical.

Even though all the meshes that were examined yield the same result for thrust and power, the wake is properly resolved only for the meshes of medium (11M) and fine (20M) resolution. Therefore, the mesh of medium resolution (11M) will be chosen for the simulations of the present thesis, since it is the coarsest mesh that is capable of resolving both the forces and the wake.

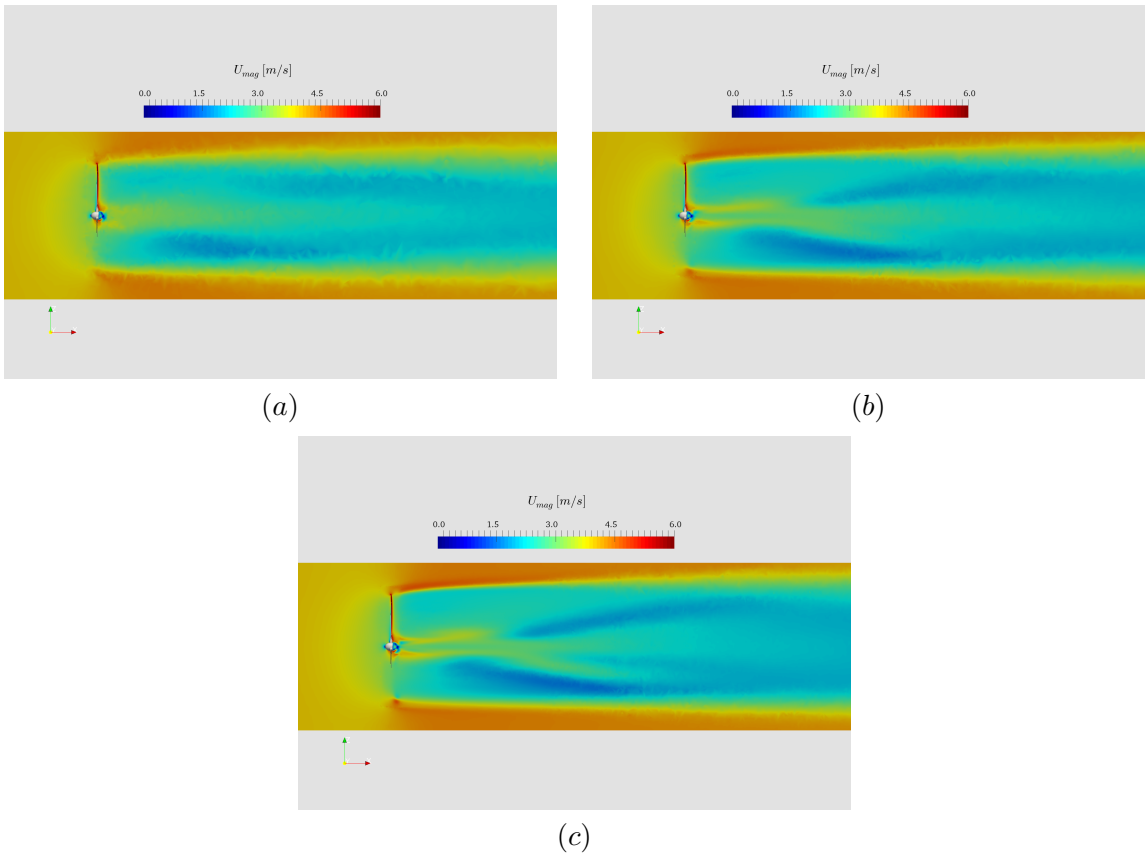


Figure 4.11: MRF steady-state simulation. Visualization of the velocity magnitude at $y = 0$ cross-section of the mesh for: (a) 8.5M, (b) 11M and (c) 20M.

4.5 Induction computation method

Another step that has to be done in the present thesis is to compute the induction of the simulated rotor. After running a CFD simulation, common physical variables like velocity, pressure and vorticity are reported as results. However, in order to assess the aerodynamics of a wind turbine it is important to be able report the induction factor of the rotor. Using the induction factor it will be possible to answer whether momentum theory is valid for a surging rotor and observe whether for extreme surge conditions induction factor becomes negative signifying a propeller state condition.

Having said that, induction factor calculation using CFD data as input is not a trivial process. In fact, there is much ambiguity on what is the right process to produce a reliable induction field out of the CFD velocity field. In the work of Rahimi et al [50], various methodologies to compute the induction factor and the angle of attack for various spanwise locations of the blade are reviewed. As discussed in this work, the AoA is not just the angle measured from CFD data for a probe point in front of the blade's airfoil. In a real flow scenario, the trailing vortices originating from variations of the blade's bound circulation vary the AoA.

Thus, in order to acquire the AoA, and thus the induced velocities, the impact of the airfoil's bound circulation needs to be removed. Various methods to remove the bound circulation have been discussed in this paper, like inverse BEM [51], the Shen models [52] and the average azimuthal technique (AAT) [53]. In the present thesis, a method similar to the Shen 2 model [54] introduced by Ferreira and Micallef will be employed for the computation of the induction factor.

4.5.1 Ferreira-Micallef induction computation method

In Ferreira-Micallef method, the flow around the blade section is assumed to be 2D, incompressible and irrotational. Furthermore, the velocity at a random point in the mesh is considered to be split in: a) the velocity induced by free vorticity, b) the velocity induced by the blade section bound circulation and c) the freestream wind velocity. The goal is basically to compute the freestream wind velocity part after removing the effect of the bound circulation in the velocity data acquired by CFD. After acquiring the freestream wind velocity a typical velocity triangle can be used to compute the induction factor, as shown in Figure 4.12.

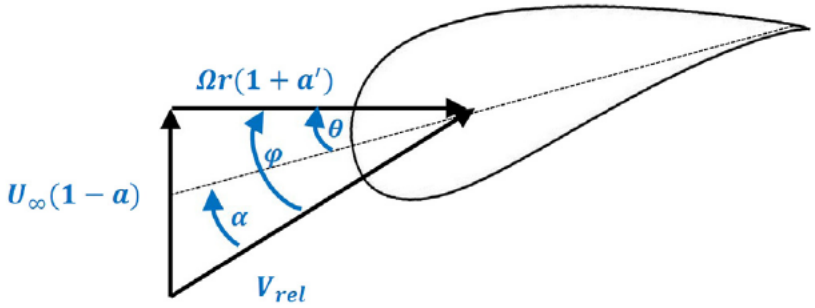
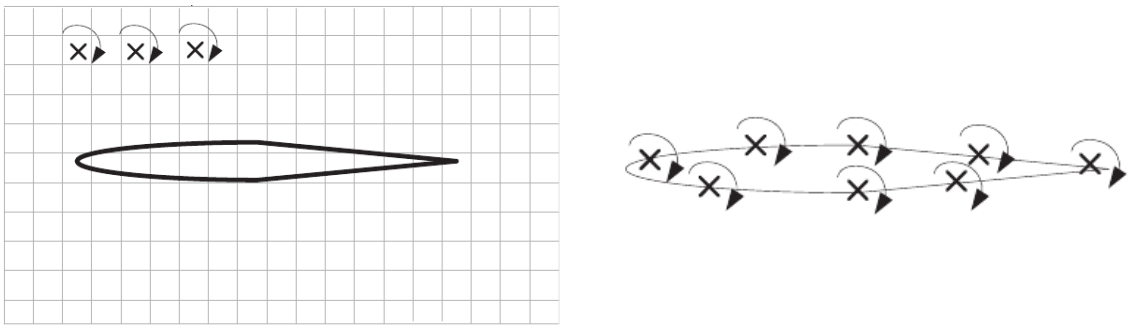


Figure 4.12: Schematic of the velocity triangle at a blade section. Taken from [50]

Both the free vorticity and the airfoil bound vorticity are approximated by Rankine point vortices with strength Γ, as shown in the schematics of Figure 4.13.



(a) (b)

Figure 4.13: Schematic of a) point vortices located at all the grid cell centers for the representation of free vorticity and b) point vortices located at the airfoil representing the bound vorticity. Schematics were taken from [50].

Control points, i.e. points where the calculation of the equations will be done, are positioned around the blade's airfoil in a form of a spiral, as shown in Figure 4.14 . The velocity induced at a control point by a Rankine point vortex located in (x_ω, y_ω) is then computed based on:

$$u_\omega = \frac{\Gamma}{2\pi} \frac{y - y_\omega}{(x - x_\omega)^2 + (y - y_\omega)^2} \quad , \tag{4.3}$$

$$v_\omega = -\frac{\Gamma}{2\pi} \frac{x - x_\omega}{(x - x_\omega)^2 + (y - y_\omega)^2} \quad , \tag{4.4}$$

where $\Gamma = \omega_z dx dy$ and ω_z is the vorticity given by CFD data in the direction normal to the plane of the blade section.

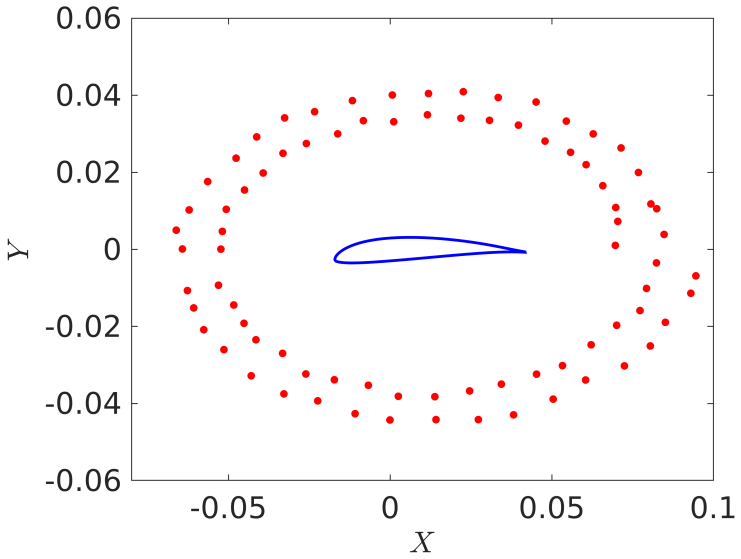


Figure 4.14: Control points positioned in a spiral around the blade section

For each control point the velocity (u_i, v_i) can then be decomposed as follows:

$$u_i = \sum u_\omega + U + \sum u_b \quad , \tag{4.5}$$

$$v_i = \sum v_\omega + V + \sum v_b \quad , \tag{4.6}$$

where $\sum u_\omega$ and $\sum v_\omega$ are the induced velocities by free vorticity, $\sum u_b$ and $\sum v_b$ are the induced velocities by the bound vorticity of the airfoil and (U, V) is the local freestream wind velocity at the quarter-chord of the blade section. Using a python script inside Paraview and least-square method to solve this system of equations, U and V are obtained which can then be used to compute the induction factors based on the velocity triangle of Figure 4.12. Specifically, the process that was followed in the present thesis to compute the induction can be broken down in the following steps:

1. Import the decomposed CFD data of openfoam inside paraview using the decomposed reader.
2. Build a paraview tree and record a macro while building it. Some of the steps that have to be done in the tree:
   (a) Rotate the domain accordingly based on the current timestep so that one of the blades is always aligned with the z-axis.
   (b) Slice the domain in the z-plane direction in the bladewise location of choice.
   (c) Use the extract block tool to extract the points of internal mesh and the airfoil.
   (d) Use the calculator to extract the coordinates of both internal and airfoil points.
   (e) Use cellsize and cellcenter tools to compute the areas of all the 2d slice cells and the cell center locations, respectively.
3. Open the python script created by the macro of paraview and use it as a basis to create a new python script where the calculation of induction will be performed for every timestep and for every spanwise location of the blade using two for-loops.

The end-result of this process is basically to open paraview press a macro button, run the script and get the computed induction in the form of text files which can then be opened by Matlab and plot the desired results. The python script that was used inside paraview can be found in the appendix.

At this point it is important to mention the main conclusions of Rahimi et al. [50] regarding the various methods of induction computation including the method of Ferreira-Micallef which was used in the present thesis. Specifically, after comparing all the different methods Rahimi et al. concluded that:

1. Even though high-fidelity CFD data are employed in all of these methods, simplified 2D theory is employed to extract the induction factors.

2. In blade regions like the root and the tip where 3D flow phenomena occur, the methods do not have a mechanism to account for chordwise bound circulation. This leads to big differences between the methods' results which deems them unreliable for these blade regions.

3. In the blade regions where 3D flow occurs (e.g. root, tip), the computed induction seems to be highly depended on the distance of the control points from the airfoil.

4. In the mid-span of the blade where 3D flow is usually not present, all methods show good agreement with each other and with experiments. Thus, they can be considered as reliable for these regions.

4.5.2 Studying the sensitivity to location and number of control points

As described in the previous subsection, the induction computation method employed in the present thesis uses some control points where the solution is computed. The control points were already graphically presented in Figure 4.14. In the present subsection, the sensitivity of the induction computation method to the location and quantity of the control points is examined. The aim of this sensitivity study is to find out what is the appropriate number of points and appropriate distance from the airfoil that should be used for the purposes of the thesis. The fixed-rotor simulation data are employed for this sensitivity study since there is no time-stepping and only one span-wise for-loop is required to extract the induction at several locations of the blade.

In Figure 4.15, the distance of the control points from the blade is examined. Four different distances from the airfoil are investigated, namely $0.1c$, $0.3c$, $0.5c$ and $0.7c$. The position of the control points for the distances of $0.1c$ and $0.7c$ can be seen in Figure 4.16 and Figure 4.17, respectively. The first thing that is observed in Figure 4.15 is that the results seem to concur with the conclusion of Rahimi et al [50] that the induction computation at the root and tip regions is highly dependent to the distance of the control points from the blade. In contrast, at the mid-span region $(40 - 60\%)$ the induction seems to be insensitive to the distance of the control points, provided that the distance is higher than $0.1c$. Based on this figure, it makes sense to choose $0.5c$ for the computations of the present thesis, since it exhibits the least difference with the other distances above $0.1c$.

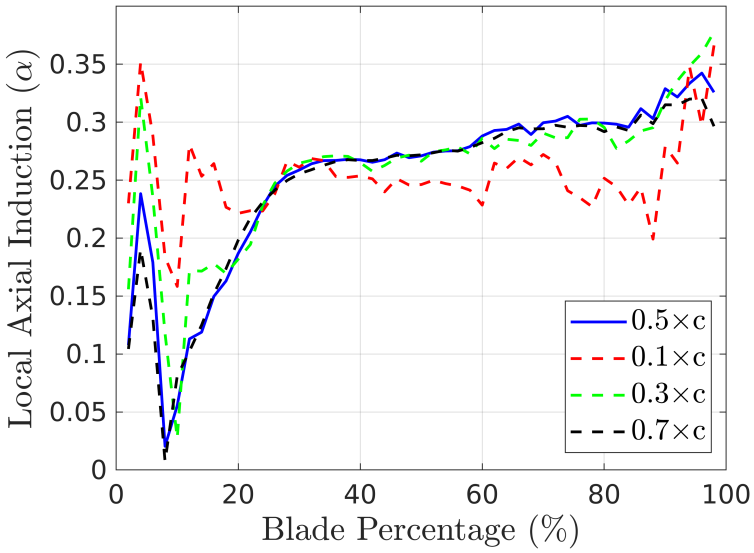


Figure 4.15: Local axial induction spanwise distribution for the fixed-rotor simulation. Assessing the sensitivity to the distance of control points from the airfoil

In Figure 4.18, the sensitivity of the induction computation method to the number of control points is examined. Three different numbers are investigated, namely 40, 80 and 160 points. The cases of 40 and 160 points can be seen in Figure 4.19 and Figure 4.20, respectively. Based on Figure 4.18, it is observed that induction computation method is insensitive to the number of the control points for all spanwise locations of the blade. Thus, 80 control points will be used for the induction computations of the present thesis.

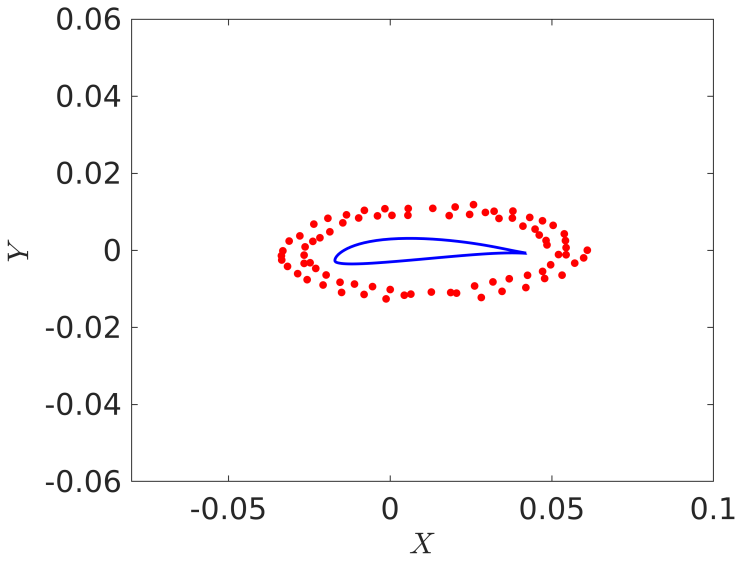


Figure 4.16: Control points located at a distance of $0.1c$ from the airfoil at a spanwise location of approximately 50% of the blade.

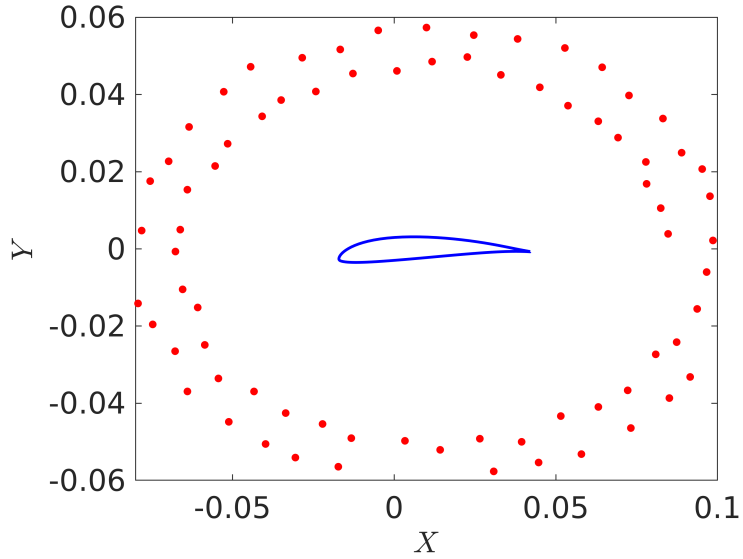


Figure 4.17: Control points located at a distance of $0.7c$ from the airfoil at a spanwise location of approximately 50% of the blade.

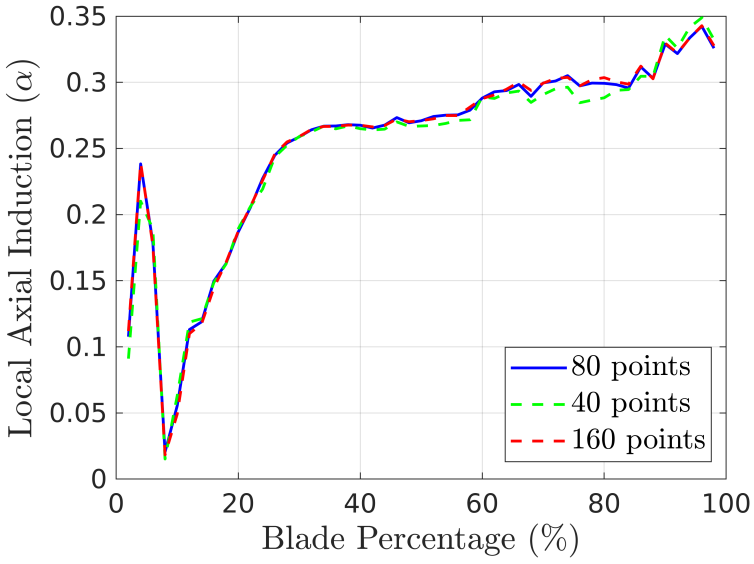


Figure 4.18: Local axial induction spanwise distribution for the fixed-rotor simulation. Assessing the sensitivity to the number of control points used

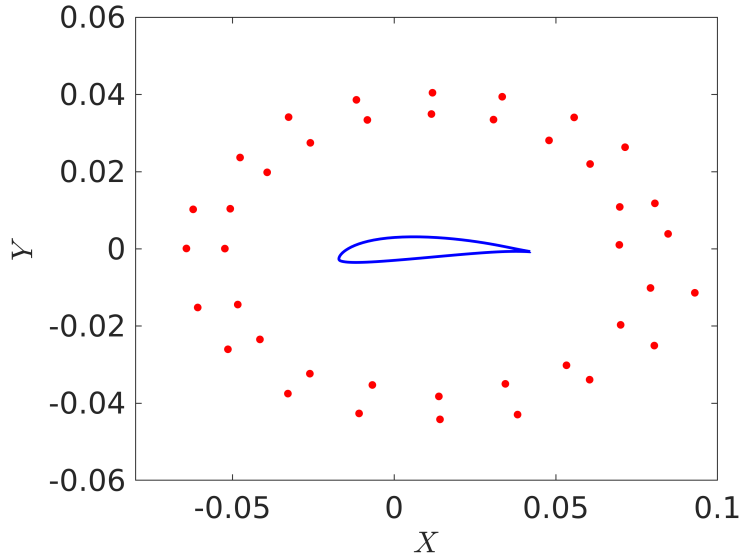


Figure 4.19: 40 control points located at a distance of $0.5c$ from the airfoil at a spanwise location of approximately 50% of the blade.

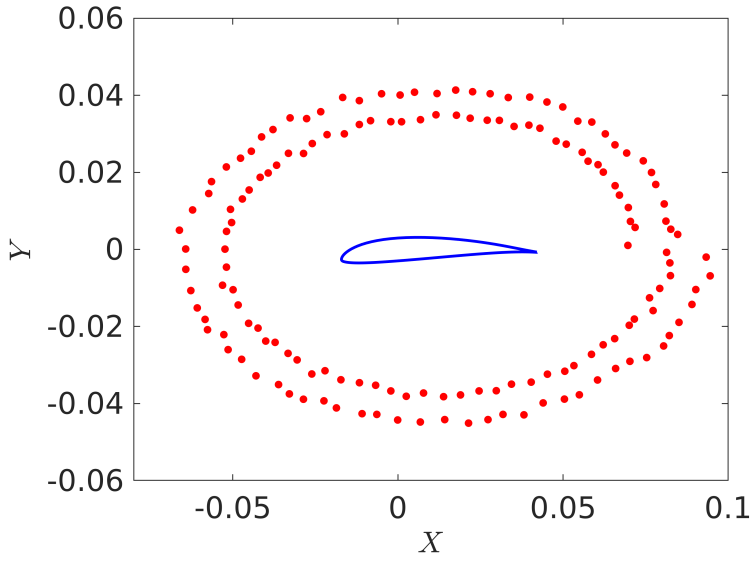


Figure 4.20: 160 control points located at a distance of $0.5c$ from the airfoil at a spanwise location of approximately 50% of the blade.

Chapter 5

CFD Model Validation

In the present section, the CFD model of the thesis is validated. It has been pointed out several times in the thesis that validating a CFD model is a very crucial step. Through the validation process, trust is built on the CFD model and the results it produces. In contrast to most works in the literature, the CFD model of the present thesis will not be validated with lower-fidelity codes (i.e. AD, BEM, AL etc.) but with high-fidelity CFD and experimental results produced in the UNAFLOW project [37].

The validation process will be broken down into two steps. First in subsection 5.1, the results of the CFD model for the fixed non-surging rotor will be compared to the fixed-turbine results of UNAFLOW. Then in subsection 5.2, the transient CFD model that incorporates the surge motion will be compared to UNAFLOW results for the surging wind turbine.

5.1 Fixed-rotor validation

In this subsection, the CFD model is validated for the case of the fixed-rotor, i.e. for the case where the UNAFLOW rotor operates under the condition #1 of Table 3.2. The 11M mesh is employed and an MRF steady-state simulation is performed. As already shown in Figures 4.9 and 4.10, the solution of the steady-state simulation converges after about 1000 iterations. This converged solution is now used as an initial condition for a transient simulation operating under the exact same conditions (i.e. #1 of Table 3.2). The only difference with the steady-state simulation is that this transient simulation employs the a sliding mesh technique to resolve the rotational motion of the blades and, thus, a difference in the computed thrust/power should be expected. It is important to validate the accuracy of the transient fixed-rotor simulation employing the sliding mesh technique since this will be the model where the surge motion will be applied.

After letting the transient simulation run for a few seconds, the solution stabilizes and the

final values of thrust and power are obtained. In Table 5.1, thrust and power computed from the CFD model are compared to the equivalent values produced in the UNAFLOW project [37]. At this point it is important to note that the UNAFLOW CFD model is highly more sophisticated than the one of the present thesis. Specifically, the main differences are that: *a*) UNAFLOW CFD model simulated the full configuration of the FOWT, i.e. rotor+nacelle+tower, while the thesis CFD model simulates only the rotor of the FOWT, and *b*) a fifth-order spatial discretization scheme was employed in UNAFLOW while only a second-order scheme was employed in the thesis CFD model. Even though the CFD model of the present thesis is a simplified model, it is observed that the it performs exceptionally good since it yields very similar results with UNAFLOW. When comparing with the high-fidelity experiment of UNAFLOW, the model yields an error of 3.75% for the thrust and 0.69% for the power. While when comparing with the highly sophisticated CFD model of UNAFLOW, the model yields exceptionally low errors of 1.88% for the thrust and 0.09% for the power.

Table 5.1: Validation of the fixed-rotor model with the results of UNAFLOW [37]. Thesis CFD model consists of a URANS simulation employing a sliding mesh technique to resolve the rotation of the rotor.

	Experiment	**UNAFLOW CFD**	**Thesis CFD**
T (N)	35.91	36.57	37.26
% Error From Experiment	—	+1.84 %	+3.75 %
% Error From UNAFLOW CFD	—	—	+1.88 %
P (W)	83.79	84.29	84.37
% Error From Experiment	—	+0.59 %	+0.69 %
% Error From UNAFLOW CFD	—	—	+0.09 %

Besides total thrust, the spanwise distribution of thrust is also compared with UNAFLOW CFD results, as shown in Figure 5.1. The spanwise thrust force of the blade was acquired using the "forces" function object that is available in OpenFoam and specifically its binData functionality to break down the blade in about 1000 sections. It is observed that the shape of the two thrust distributions is almost identical, with the CFD model of the thesis yielding slightly higher values in all spanwise locations.

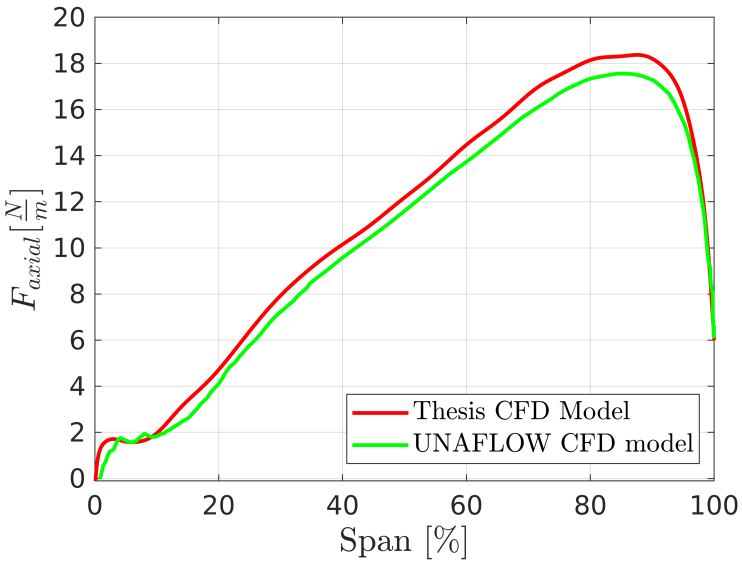


Figure 5.1: Thrust spanwise distribution along the blade. Comparison of the thesis CFD model with UNAFLOW CFD model.

5.2 Surging-rotor validation

After validating the CFD model for the fixed-rotor operating conditions, it is also crucial for the research goals of the thesis to validate the model for the surging rotor operating conditions. It is of paramount importance to build trust on the computational model that will be used to answer the research questions. In fact, constructing an accurate CFD model that would be able to assess the aerodynamics of a surging rotor was probably the first and most important step of the present thesis. Building such a high-fidelity CFD model was quite a laborious task that required several months of troubleshooting and the fact that the FOWT literature contains very few high-fidelity CFD models proves how demanding such a task was. Therefore, validating that the thesis CFD model is able to capture the loads of a surging rotor is probably even more important than answering the other research goals that were set.

The surging parameters $f_s = 2Hz$ and $A_s = 8mm$ that were examined in the UNAFLOW project [37] are also investigated in the present thesis to assess whether the CFD model is acccurately predicting the unsteady loads of a surging rotor. An unsteady simulation is performed for the operating conditions listed in row #1 of Table 3.2 and the computed thrust is shown in Figure 5.2. The thrust oscillation computed by the thesis CFD model for a period of the surge motion is compared to the UNAFLOW experiment and UNAFLOW CFD model. In order to better visualize the oscillating part of the signal, only $\Delta T = T_{surge} - T_{steady}$ is shown. By comparing just the ΔT part, we examine how well the thesis CFD model represents the dynamic effects taking place during the surge motion of the rotor. The thesis CFD model seems to have a very good agreement with the UNAFLOW high-fidelity experiment and CFD code. The frequency of the thrust oscillation computed by the thesis CFD model is almost

identical with the frequency of the thrust oscillation computed by the UNAFLOW project. However, the magnitude of the thrust oscillation seems to be a bit smaller from the one predicted by the UNAFLOW CFD and experiment.

In order to quantify this difference in magnitude, an error indicator ϵ is defined as follows:

$$\epsilon = \frac{\Delta T - \Delta T_{ref}}{2N_{ref}} \cdot 100 \quad , \tag{5.1}$$

where ΔT is the thrust oscillation of the signal for which the error indicator is computed, ΔT_{ref} is the thrust oscillation of the signal used as reference and N_{ref} is the total variation of the signal used as reference, i.e. $N_{ref} = \Delta Tref, max - \Delta Tref, min$. The error of the thesis CFD model and the UNAFLOW CFD model using as reference the UNAFLOW experiment are presented in Figure 5.3 and 5.4, respectively. For both figures the ΔT signal of the UNAFLOW experiment serves as the reference values for the error computation. In Figure 5.3, it is observed that the thesis CFD model under or over predicts the magnitude of the thrust oscillation by a value of about $10-15\%$ while the overall average error is 0.1324%. Similarly, in Figure 5.4, it is observed that the UNAFLOW CFD model under or over predicts the magnitude of the thrust oscillation by a value of also $10-15\%$ while the overall average error is -2.9035%. The conclusion that can be drawn from Figures 5.3 and 5.4 is that the thesis CFD model has the same level of error with the UNAFLOW CFD model in the prediction of the thrust oscillation magnitude. Thus, the thesis CFD model performs surprisingly well when taking into account that it aimed for a low computational cost by using only second-order spatial discretization schemes and omitting the nacelle and the tower from the analysis.

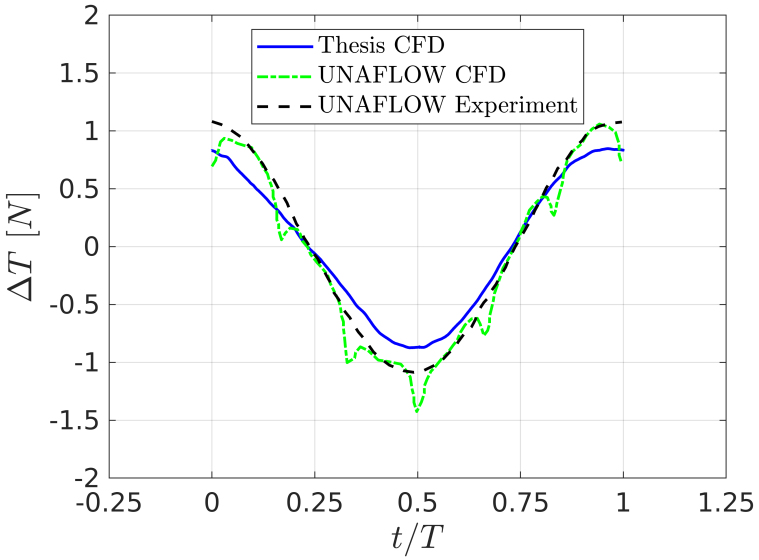


Figure 5.2: Thrust oscillation $\Delta T = T_{surge} - T_{steady}$ during a period of a surge motion with $f_s = 2Hz$ and $A_s = 8mm$ (UNAFLOW parameters). Comparison of the thesis CFD model with UNAFLOW CFD and experiment.

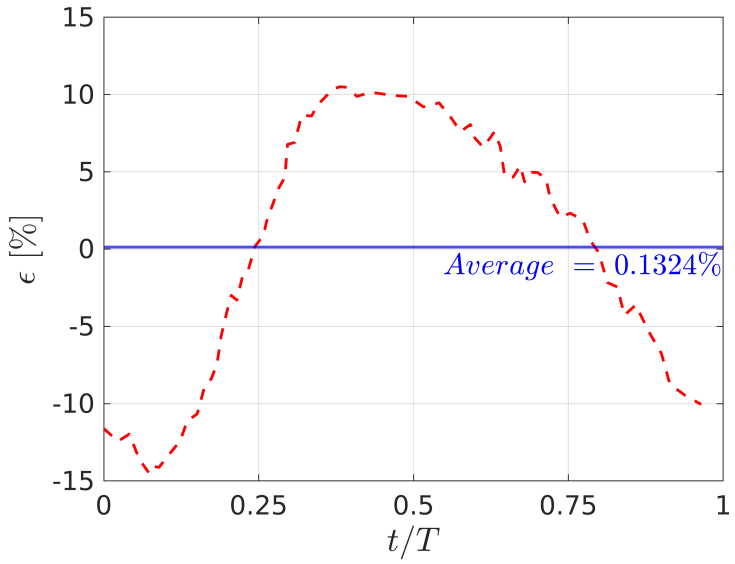


Figure 5.3: Error indicator ϵ (%) of the thesis CFD model for the thrust computation of the rotor surging at $f_s = 2Hz$ and $A_s = 8mm$ (UNAFLOW parameters). The ΔT values of the UNAFLOW experiment are used as reference.

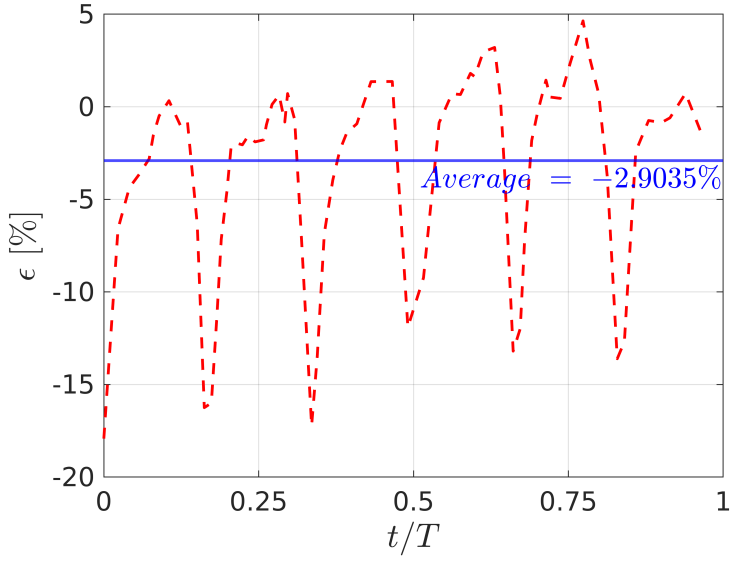


Figure 5.4: Error indicator ϵ (%) of the UNAFLOW CFD model for the thrust computation of the rotor surging at $f_s = 2Hz$ and $A_s = 8mm$ (UNAFLOW parameters). The ΔT values of the UNAFLOW experiment are used as reference.

Chapter 6

Results

In this chapter the results for the two test cases of table 3.2 will be presented. Case #1 corresponds to the surge parameters examined in the UNAFLOW project and represent a mild sea state. This case was examined to validate the CFD model with UNAFLOW experimental & CFD data, but also to check whether momentum theory is able to capture the induction for such a mild surge motion. Case #2 corresponds to a rougher sea state with surge parameters leading to $V_{max,red} > 1$. As discussed in chapter 4, such a big $V_{max,red}$ leads to large C_T oscilations which, according to some authors, may cause propeller or vortex ring state. For both cases, the induction of the surging rotor is acquired using Ferreira-Micallef model which was explained in section 4.5.1.

6.1 Results for case #1 with $V_{max,red} = 0.025$

In this section, the results of case #1 are presented. The displacement and velocity of the surging rotor are shown in Figure 6.1 for one surge period. The oscillation of the total thrust of the rotor has already been shown in Figure 5.2 where the CFD model of the present thesis appeared to be very close to high-fidelity experimental data. In Figure 6.2, the thrust force coefficient of the surging rotor is shown. It is observed that C_T follows the inverse shape of U_{surge}. This was expected since the velocity experienced by the rotor is $V - V_s - V_{in}$, as shown in Eq. 2.25.

Before moving on the induction plots, it is also important to visualize the local thrust coefficient of the rotor for root, tip and mid-span regions. In Figure 6.3, the local thrust coefficient is plotted during one surge period for 5, 50 and 95% of the blade. It is observed that, besides the surge frequency, other frequencies are also present in the root and tip regions of the blade. This could be explained by considering that the root and tip regions have strong trailing vortices in the wake. These large vortical structures affect the V_{in} part of Eq. 2.25. In fact, the shape of the C_T distributions for the root and tip indicate that there are several

vortical structures present near the root and probably one large vortical structure near the tip.

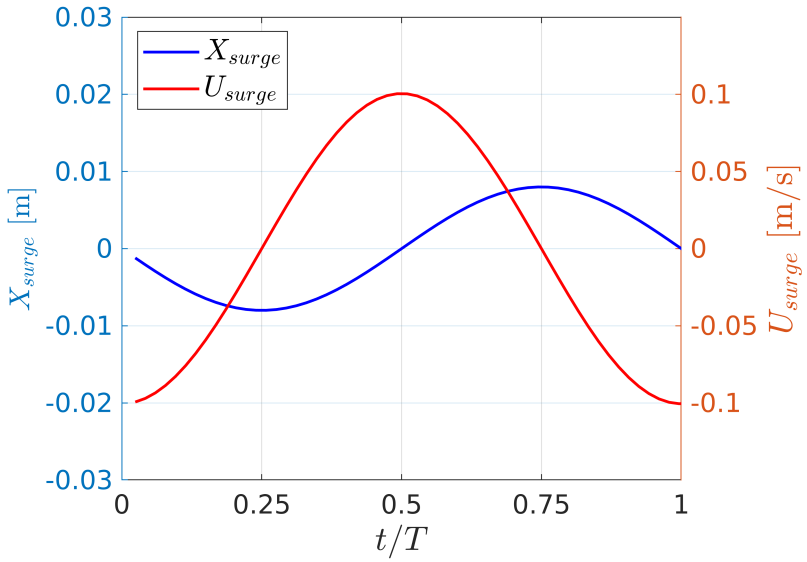


Figure 6.1: Displacement and velocity of the surge motion of the rotor for case #1

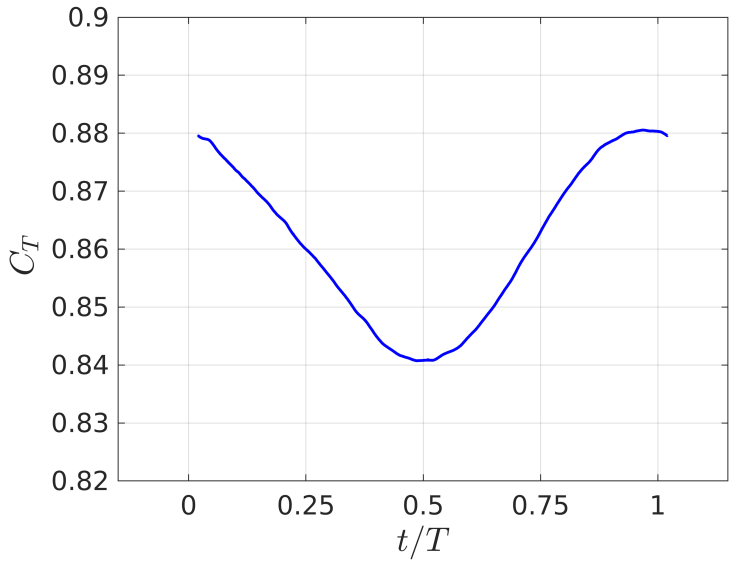


Figure 6.2: Thrust coefficient C_T vs time for the whole rotor.

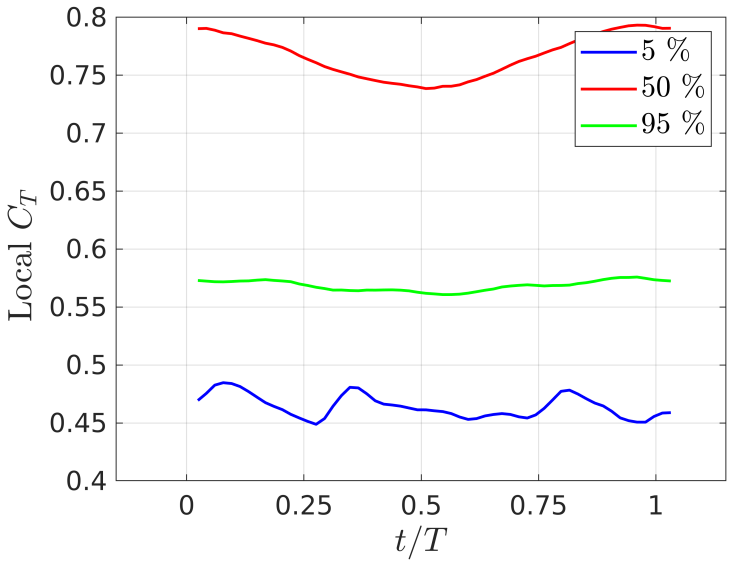


Figure 6.3: Local C_T vs time for root,tip and mid-span.

The Ferreira-Micallef model is now employed to compute the local induction of the rotor for several spanwise locations of the blade. At this point, it is important to repeat what was discussed in section 4.5.1. Same with all induction computation models discussed in the work of Rahimi et al [50], Ferreira-Micallef model makes use of 3D CFD data but it is based on 2D theory, and thus it is expected to be less accurate for tip and root regions where strong 3D flow phenomena occur. In the mid-span of the blade, 3D flow is usually not present and thus the model can be more reliable.

In Figure 6.4, the local induction computed from Ferreira-Micallef model using CFD data is compared with the induction predicted by momentum theory. To get the curve of momentum theory, the local C_T at the mid-span was used and Glauert and Prandtl corrections were applied. It is observed that the two curves seem to have similar shape but momentum theory is predicting a larger magnitude for the induction oscillation. In Figure 6.5, it is observed that the percentile difference of momentum theory to the thesis CFD model is ranging between 2 and 10% with an average of 4.5%. Thus, momentum theory seems to deliver very similar results to the ones of the thesis CFD model.

Even though Ferreira-Micallef model may not be very accurate for tip and root regions, local induction plots and the respective percentile difference of momentum theory are presented in Figures 6.6-6.9. In Figure 6.6, it is observed that momentum theory and CFD induction curves for the root region have a similar shape but momentum theory predicts a smaller magnitude of the mean induction and the induction oscillation. Similarly, in Figure 6.8, the same observation can be made for the tip region induction curves. In both root and tip regions, momentum theory is exhibiting a percentile difference of the order of 30% from the thesis CFD model.

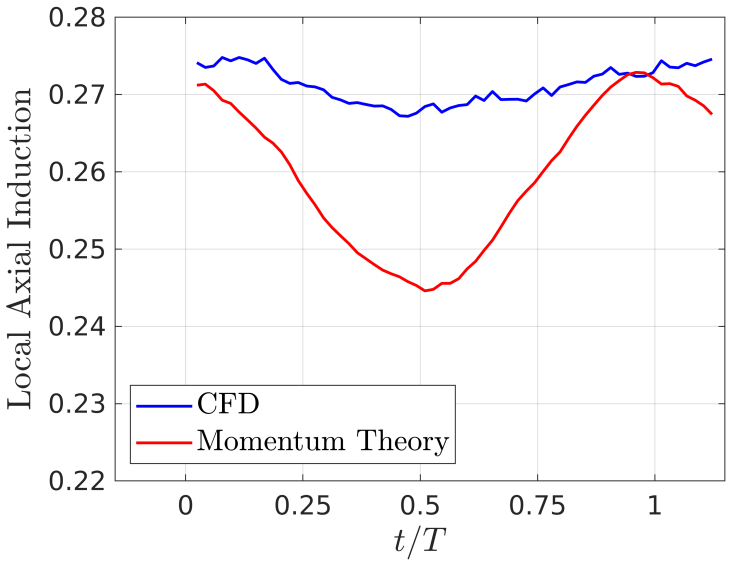


Figure 6.4: Local axial induction at mid-span as computed by CFD (Ferreira-Micallef model) and from momentum theory.

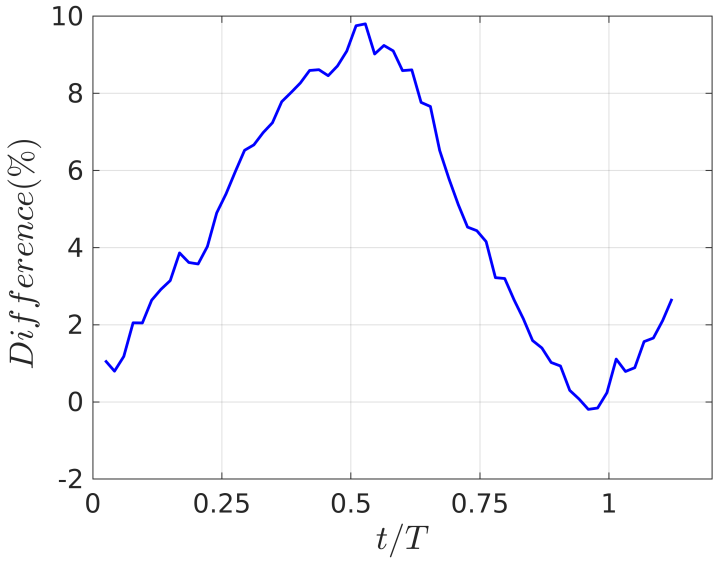


Figure 6.5: Percentile difference of momentum theory axial induction prediction when compared to CFD (Ferreira-Micallef model) prediction for the mid-span region.

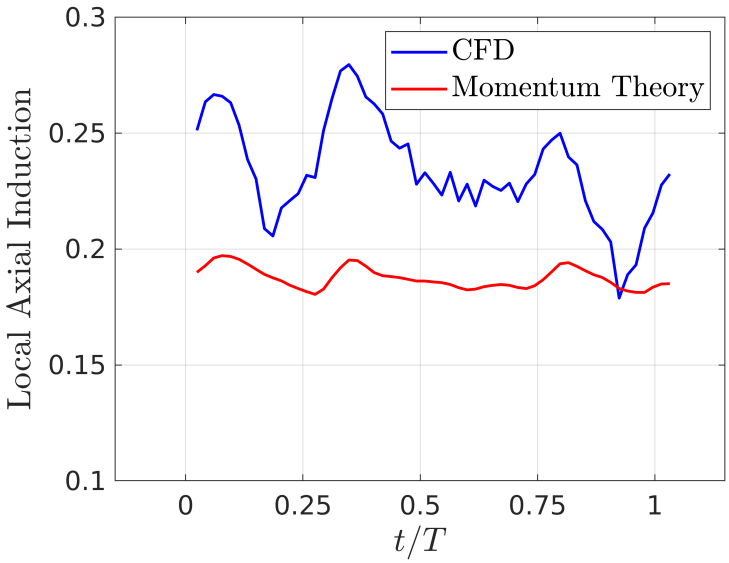


Figure 6.6: Local axial induction at the root of the blade as computed by CFD (Ferreira-Micallef model) and from momentum theory.

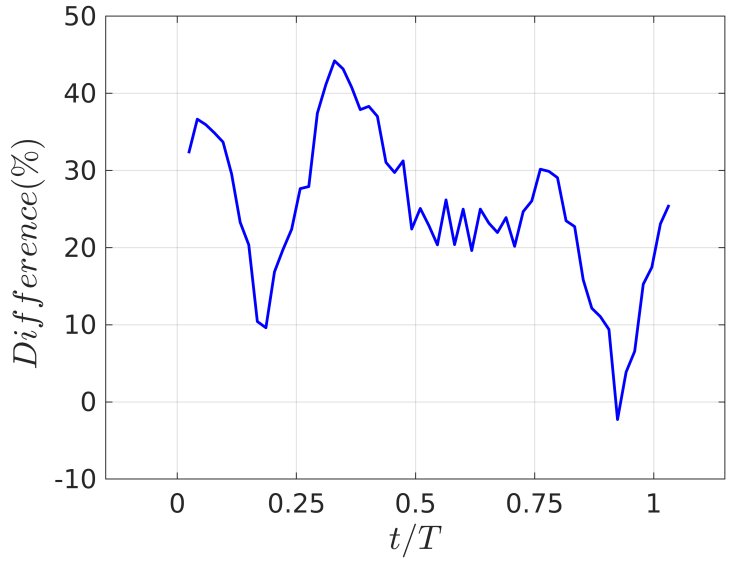


Figure 6.7: Percentile difference of momentum theory axial induction prediction when compared to CFD (Ferreira-Micallef model) prediction for the root region.

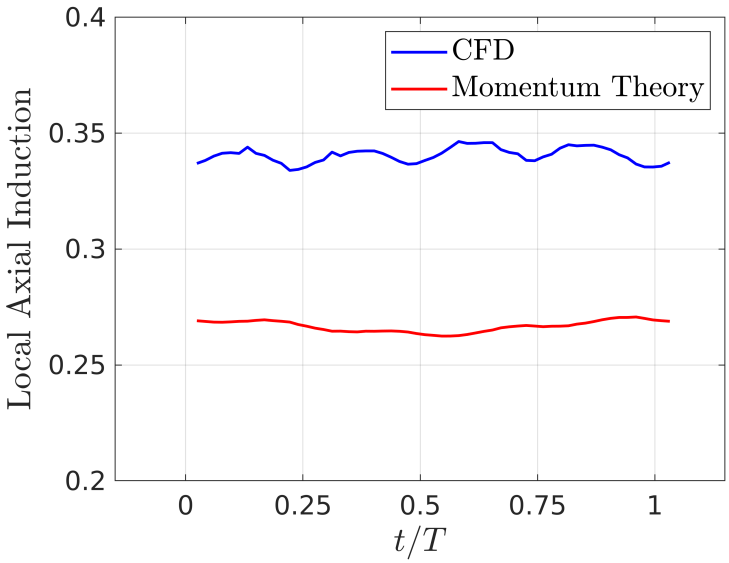


Figure 6.8: Local axial induction at the tip of the blade as computed by CFD (Ferreira-Micallef model) and from momentum theory.

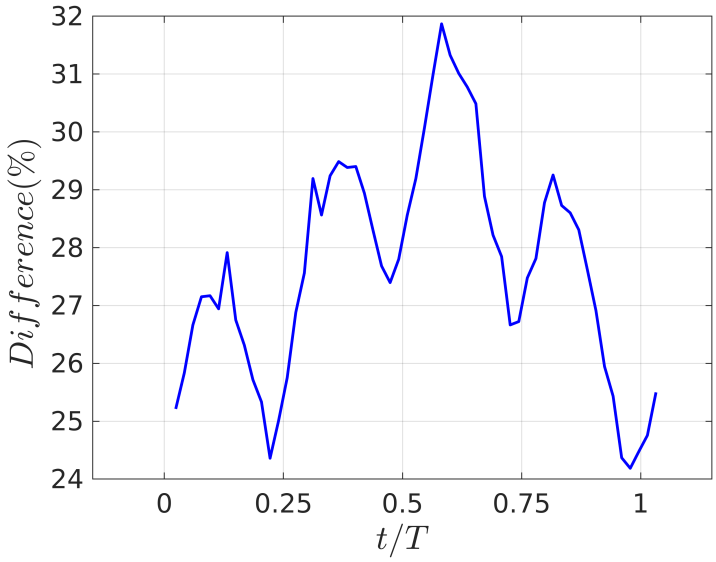


Figure 6.9: Percentile difference of momentum theory axial induction prediction when compared to CFD (Ferreira-Micallef model) prediction for the tip region.

In the present case, $V_{max,red}$ was quite low and thus the resulting oscillation of C_T was also quite small, as shown in Figure 6.2. Since C_T did not reach negative values or values above 1, induction should also not reach regimes where propeller or vortex ring states will occur and, thus, no further investigation for these states will be done for this case.

6.2 Results for case #2 with $V_{max,red} = 1.036$

In this section, the results of case #2 are presented. The aim of case #2 is to provide clarity on whether propeller or vortex ring state occur during the operation of a FOWT. Such states can occur in rough sea wave states where the surge frequency and amplitude lead to high C_T oscillations. As discussed previously, $V_{max,red}$ is a non-dimensiolized parameter which gives an indication of how much C_T will oscillate. By examining a case where $V_{max,red} = 1.036$, we make sure that at some point of the surge period the velocity of the rotor will be higher than V_{inf} and thus the rotor will experience a negative relative velocity, as shown in Eq. 2.25.

In Figure 6.10, the displacement and velocity of the surge motion are presented. It is shown that V_{surge} becomes indeed larger than $V_{inf} = 4m/s$ once in every surge period. At this point where V_{surge} becomes larger than V_{inf}, $V_{max,red}$ becomes larger than 1 and C_T is expected to acquire negative values. In Figure 6.11, the oscillation of the total C_T of the rotor is presented. It is observed that indeed C_T acquires negative values once at every surge period and specifically at the region where $V_{surge} > V_{inf}$.

Based on typical streamtube wake state figures, like Figure 2.6, some authors believe that a negative value of C_T during the operation of FOWT means that the rotor experiences propeller state. However, other authors like Ferreira et al [28] argue that negative rotor thrust does not necessarily mean that the streamtube enters propeller state because it does not have the time to adjust to the negative loading of the rotor. This is a region where not many studies have been done. Thus, the present thesis, using the developed blade-resolved CFD model, aims to provide clarity on this controversial topic.

In Figure 2.6, it is also shown that close to the region where $a = 1$ the flow is expected find a sudden stop, large vortical structures are expected to form and then advected downstream. In the present case such a phenomenon is supposed to occur when the turbine moves with a large velocity in the downstream direction and enters its own wake, i.e. in the time instant when V_{surge} becomes maximum in the positive values. In this instant the effective wind speed experienced by the rotor may drop significantly, induction may become equal to 1 and a vortex ring may be created.

To provide clarity on the topic of propeller and vortex ring states, induction plots as well as 3D visualizations of the field are performed. Similar to case #1, induction is computed using Ferreira-Micallef induction computation model. Three-dimensional visualization of the field is done using paraview. Specifically, scalar slices, streamlines and q-criterion scenes are produced to better visualize the phenomena of importance.

In Figure 6.12, the local thrust coefficient is plotted during one and a half surge period for 5, 50 and 95% of the blade. In contrast to case #1, it is observed that the surge frequency is the dominant frequency in all three spanwise positions. This difference between case #1 and case #2 should be expected since case #2 has a much higher $V_{max,red}$ and the influence of surge motion in C_T should be much larger in all spanwise locations. The oscillation of C_T in root, tip and mid-span seems to have a different magnitude with the one in the mid-span being the higher and leading to negative C_T values.

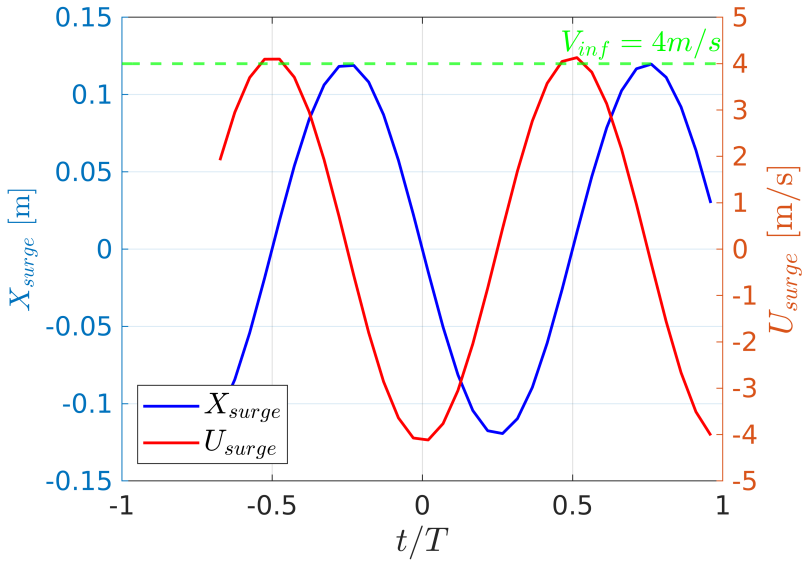


Figure 6.10: Displacement and velocity of the surge motion of the rotor for case #2

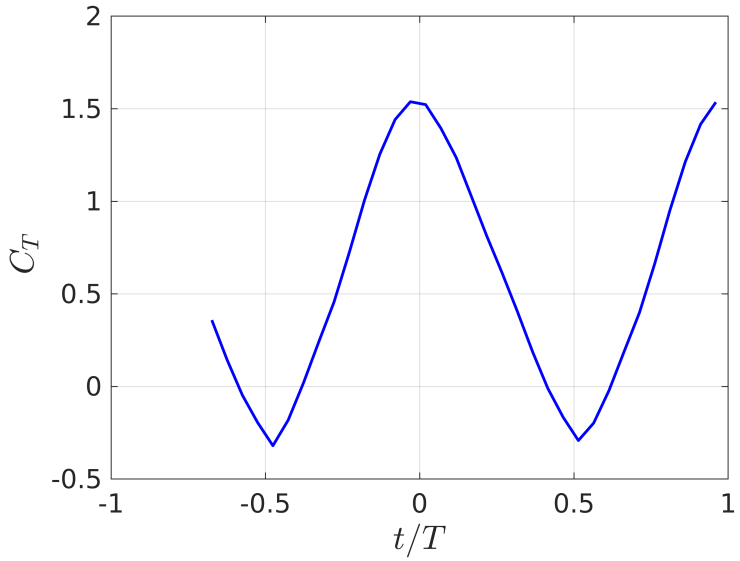


Figure 6.11: Thrust coefficient C_T vs time for the whole rotor.

In Figure 6.13, the local induction computed from CFD (Ferreira-Micallef model) and the one predicted by momentum theory are presented for the mid-span of the blade. Similarly to case #1, to get the curve of momentum theory, the local CT at the mid-span was used and Glauert and Prandtl corrections were applied. In contrast to case #1, it is observed that the two induction curves do not have a similar shape. The CFD curve seems to have a different frequency and a much different magnitude of induction oscillation. In fact, while momentum theory seems to be predicting negative induction and, thus, propeller state close

to $t/T = 0.5$, CFD seems to predict only positive values. Furthermore, momentum theory seems to be predicting values of induction larger than 1 and, thus, vortex ring state, while CFD exhibits a maximum value 0.5.

In Figure 6.14, the percentile difference of momentum theory for the mid-span induction is presented to better quantify its difference from CFD (Ferreira-Micallef model). It is observed that the difference of momentum theory is ranging between -200 and 200% with an average of -20%. Thus, momentum theory delivers wildly different results with CFD for mid-span induction. The difference of induction predicted by CFD and momentum theory could be because of what Ferreira et al [28] also suggested, i.e. the streamtube may not have the time to adjust to the sudden C_T oscillations.

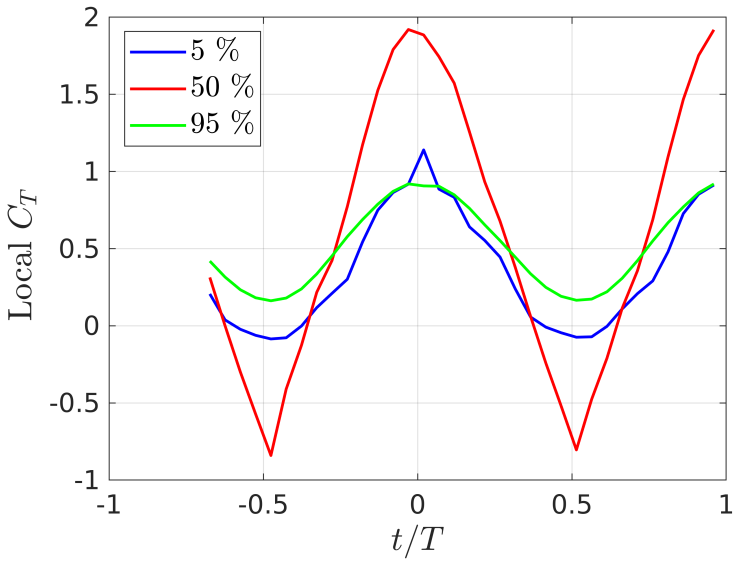


Figure 6.12: Local thrust coefficient C_T vs time for mid-span, root and tip regions of the blade.

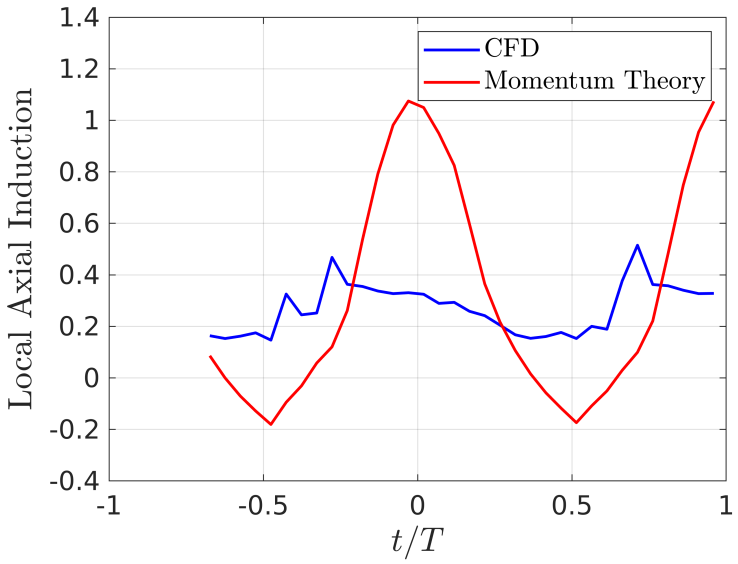


Figure 6.13: Local axial induction at mid-span as computed by CFD (Ferreira-Micallef model) and from momentum theory.

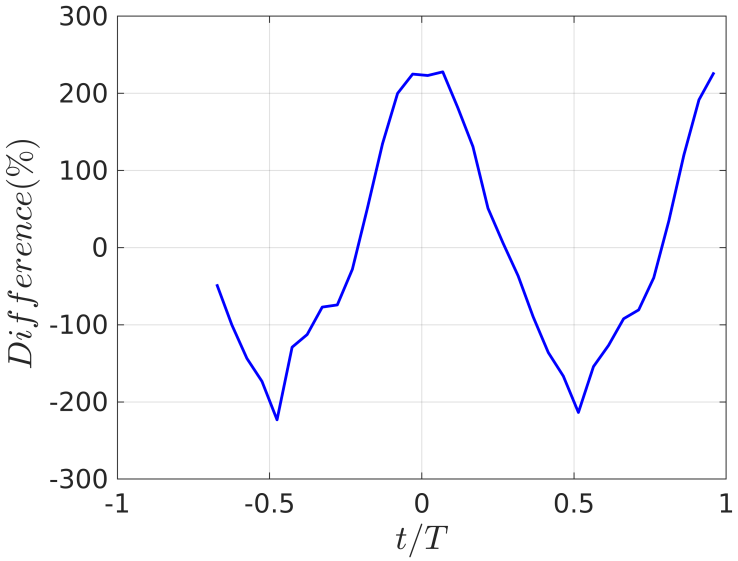


Figure 6.14: Percentile difference of momentum theory axial induction prediction when compared to CFD (Ferreira-Micallef model) prediction for the mid-span region.

Even though Ferreira-Micallef model is not expected to be very accurate for tip and root regions [50], local induction plots and the respective percentile difference of momentum theory are presented in Figures 6.15-6.18. In Figure 6.15, it is observed that momentum theory and CFD induction curves for the root region have a wildly different shape. Both momentum theory as well as Ferreira-Micallef model is not expected to perform well in the root region so

there is no clear way to identify which is the true induction curve for this region. In contrast to the root region, the induction curves predicted for the tip region by Ferreira-Micalled model and momentum theory seem to have a somewhat similar shape/frequency with a different phase and a different magnitude, as shown in in Figure 6.17. This could be attributed to the fact that less radial flow will be present at this region since there is a high rotational tip velocity which probably dominates the flow.

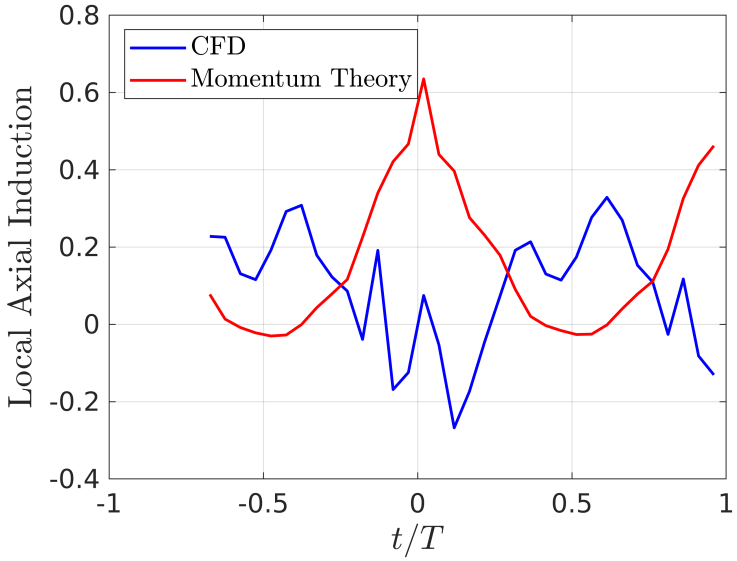


Figure 6.15: Local axial induction at root region of the blade as computed by CFD (Ferreira-Micallef model) and from momentum theory.

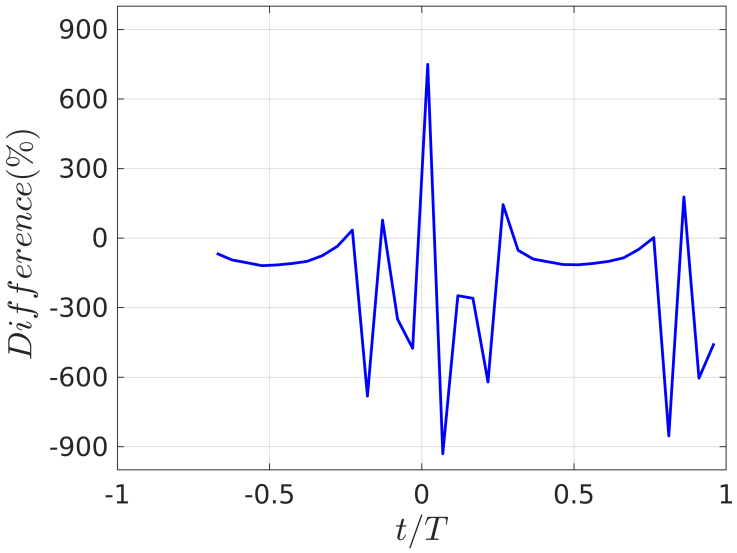


Figure 6.16: Percentile difference of momentum theory axial induction prediction when compared to CFD (Ferreira-Micallef model) prediction for the root region.

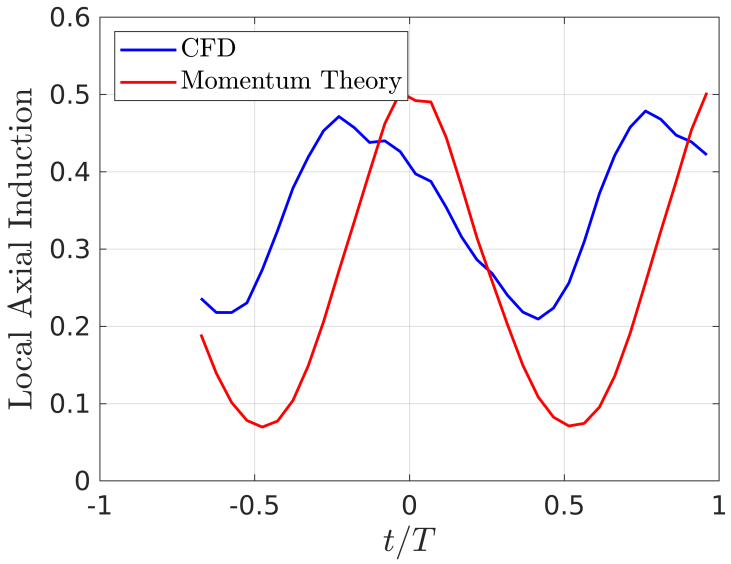


Figure 6.17: Local axial induction at tip region of the blade as computed by CFD (Ferreira-Micallef model) and from momentum theory.

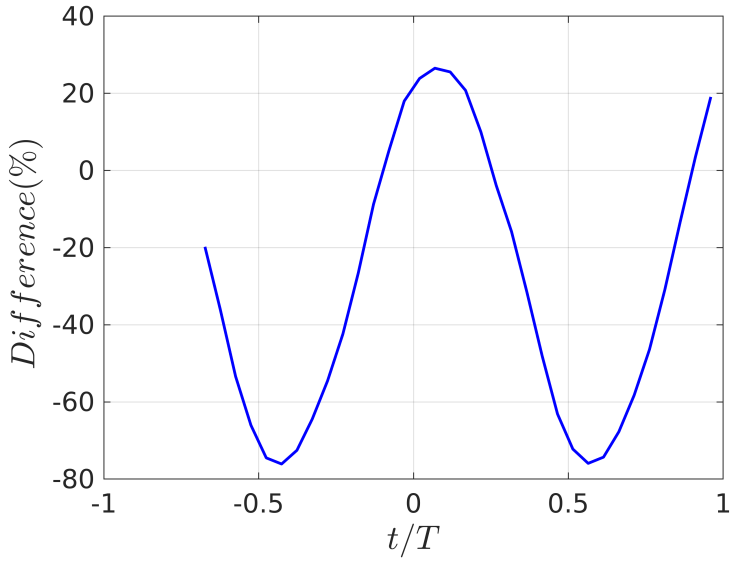
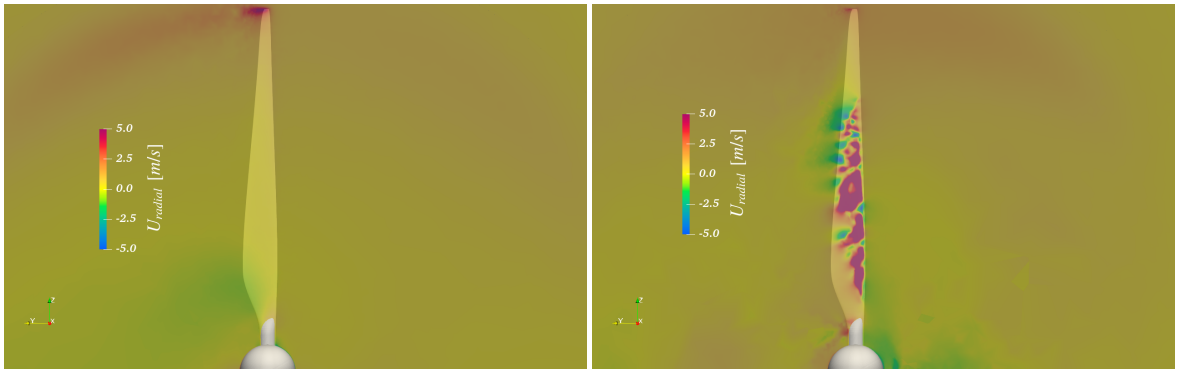


Figure 6.18: Percentile difference of momentum theory axial induction prediction when compared to CFD (Ferreira-Micallef model) prediction for the tip region.

Ferreira-Micallef model was employed in case #2 to compute the induction in several spanwise locations of the blade using CFD field data. However, someone could argue that the results of this model are not very reliable for the a rotor surging in such harsh conditions. As discussed in section 4.5.1, Ferreira-Micallef model relies on 2D theory, i.e. relies in the fact that no major radial 3D flow will take place. In case #1 the intensity of the surge velocity

fluctuations was much smaller than the one observed in case #2. Thus, someone could argue that the flow in case #2 will experience a sudden blockage and, thus, radial flow will be much more evident than in case #1. This argument can be visualized in Figure 6.19, where almost no radial flow is observed for case #1 while significant radial flow is observed for case #2. Furthermore, in Figure 6.19(b), it can be seen that no radial flow is present close to the tip region of the blade, confirming the above-mentioned argument that Ferreira-Micalled model and momentum theory produced similar induction curves in Figure 6.17 shapes due to the absence of radial flow.

In the work of Rahimi et al [50], several induction calculation models together with Ferreira-Micallef model were deemed as unreliable for root and tip regions since they delivered different results. On the other hand, in the mid-span region all the induction calculation models were deemed as reliable since they delivered identical results and no radial flow is generally expected in this region. This was explained in the start of this chapter and only the induction results for the mid-span region would be used to draw conclusions. However, in the case of the present thesis, radial flow seems to be present in the mid-span of the rotor surging in the harsh surge conditions of case #2. Thus, it can be concluded that the induction curve of Figure 6.13 can not be trusted to draw reliable conclusions on whether propeller or vortex ring state occurs.



(a) (b)

Figure 6.19: Radial flow magnitude for a) case #1 b) case #2 at the time where U_{surge} is maximum.

6.2.1 3D visualisations of the CFD flow field

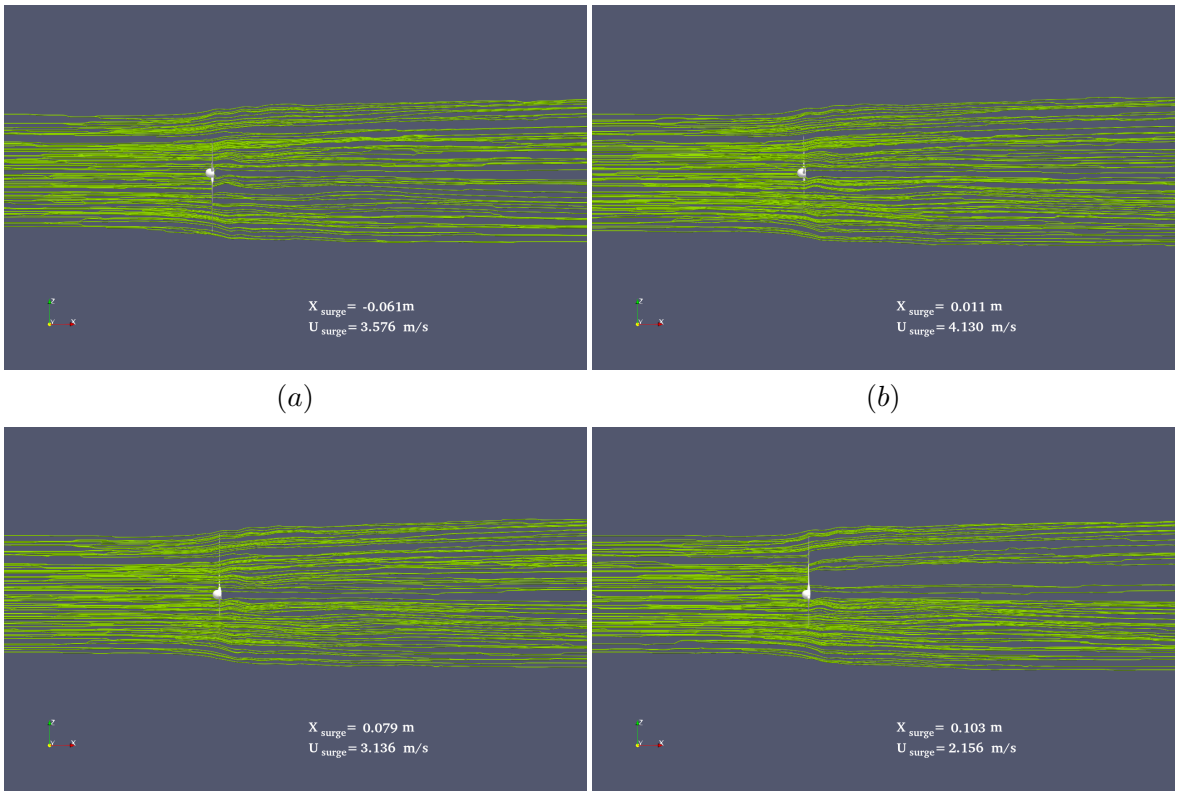
Fortunately, besides producing induction plots, 3D visualisations of the CFD flow field were also computed to draw conclusions on whether propeller or vortex ring states occur. In Figures 6.20-6.23, three streamline scenes and one velocity contour scene are presented to provide clarity on whether propeller state occurs. In Figures 6.24-6.26, two vorticity contour scenes and one q-criterion isosurface scene are presented to visualize whether vortex ring state occurs. In all of these figures four time instances are shown, namely $\frac{t}{T} = 0.41$, $\frac{t}{T} = 0.5$, $\frac{t}{T} = 0.61$ and $\frac{t}{T} = 0.66$. As shown in Figures 6.10 and 6.11, at $\frac{t}{T} = 0.5$ two things are happening:

1. The rotor starts moving backwards (positive X_{surge}), hits on its own wake, and surge velocity reaches its maximum value. In this instant the effective wind speed experienced by the rotor may drop significantly, induction may instantly become equal to 1 and a vortex ring may be created.

2. The thrust coefficient, C_T, acquires negative values. Negative values of C_T could lead to negative induction values and, thus, propeller state.

6.2.1.1 Scenes for the investigation of propeller state

In Figures 6.20 and 6.21, the streamlines on the side (y-plane) and top (z-plane) views of the rotor are shown. 2000 streamline seed points are positioned $1R = 1.18m$ upwind of the rotor and have a circular layout with a radius of $1R$. With this streamline setup the streamtube upwind and downwind of the rotor is visualized. It is observed that the streamtube is always expanding, and thus it is in windmill state, even though C_T acquires negative values. In Figure 6.22, a contour of x-component (axial) of the velocity is also shown on the side-view of the rotor. This scene exhibits, again, that the streamtube of the rotor is expanding since the cells downwind of the rotor that exhibit a velocity lower than U_{inf} are forming an expanding shape. These results boosts the argument of Ferreira et al [28] that the flow in the streamtube does not have the time to adjust to the negative loading of the rotor and remains in wind-mill state (positive induction values) throughout the surge motion.

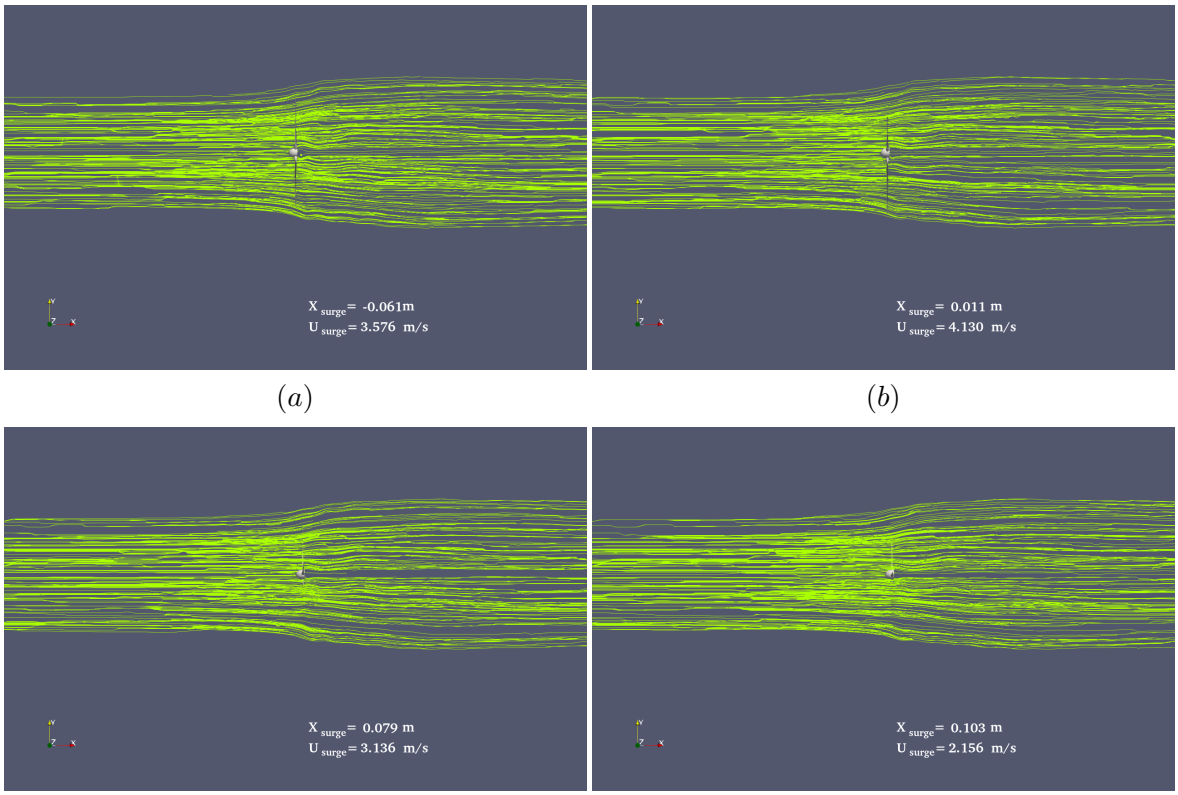


(a) (b)

(c) (d)

Figure 6.20: Streamlines on y-plane, i.e. on the side-view of the rotor. Four time instances are shown: (a): $\frac{t}{T} = 0.41$ (b): $\frac{t}{T} = 0.5$ (c): $\frac{t}{T} = 0.61$ (d): $\frac{t}{T} = 0.66$.

Close to $\frac{t}{T} = 0.5$, V_{surge} becomes larger than V_{inf} and the relative velocity experienced by the rotor becomes negative, as shown in Eq. 2.25. Negative values of relative velocity means that the flow is travelling from the back to the front of the rotor. Thus, someone could argue that in Figures 6.20(b) and 6.21(b), since the flow is travelling from right to left, the streamtube is contracting and hence it is in propeller state. This logic, however, is erroneous since the streamlines of Figures 6.20 and 6.21 are produced with the inertial reference frame velocity and not with the relative reference frame velocity. In Figure 6.23, the streamlines produced based on the relative reference frame velocity are shown. It is observed, once again, that the streamtube is expanding but this time from right to left since the velocity is negative.



(a) (b)

(c) (d)

Figure 6.21: Streamlines on z-plane, i.e. on the top-view of the rotor. Four time instances are shown: (a): $\frac{t}{T} = 0.41$ (b): $\frac{t}{T} = 0.5$ (c): $\frac{t}{T} = 0.61$ (d): $\frac{t}{T} = 0.66$.

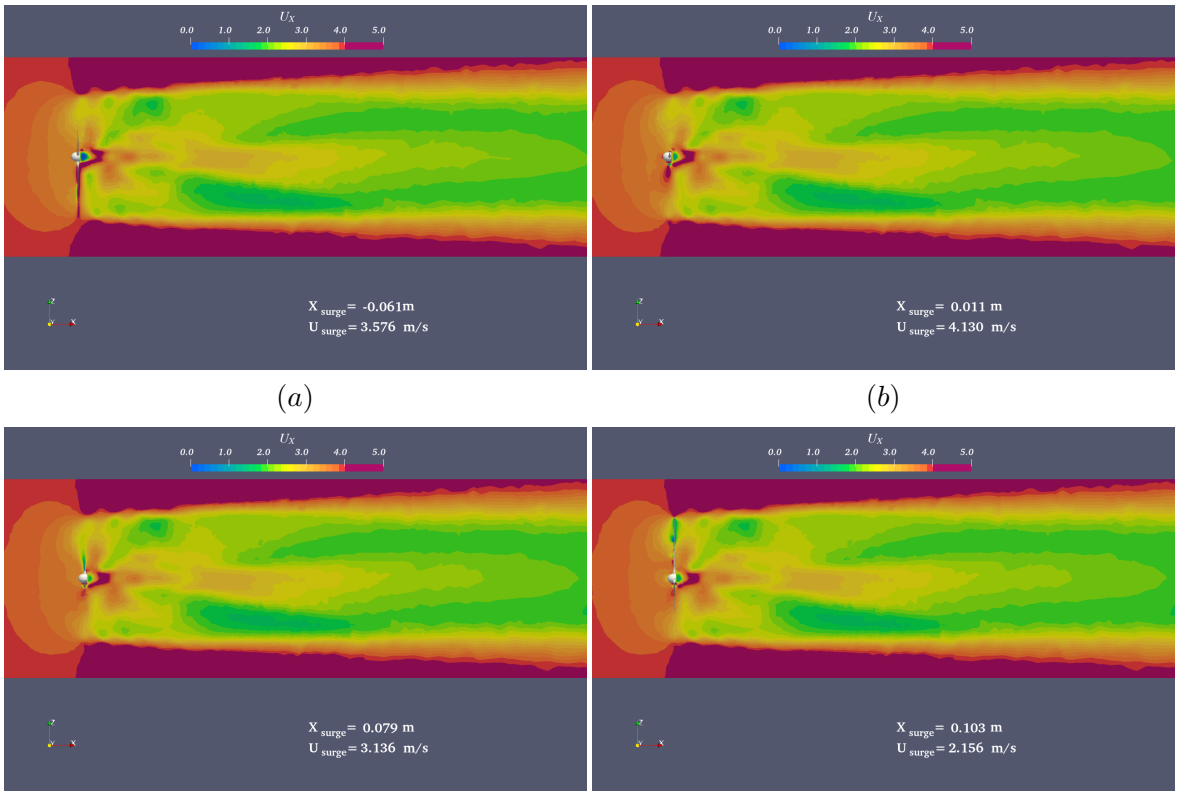


Figure 6.22: Contour of U_X on y-plane, i.e. on the side-view of the rotor. Four time instances are shown: (a): $\frac{t}{T} = 0.41$ (b): $\frac{t}{T} = 0.5$ (c): $\frac{t}{T} = 0.61$ (d): $\frac{t}{T} = 0.66$.. Velocities higher than $U_{inf} = 4\ m/s$ are colored with the same dark-red color.

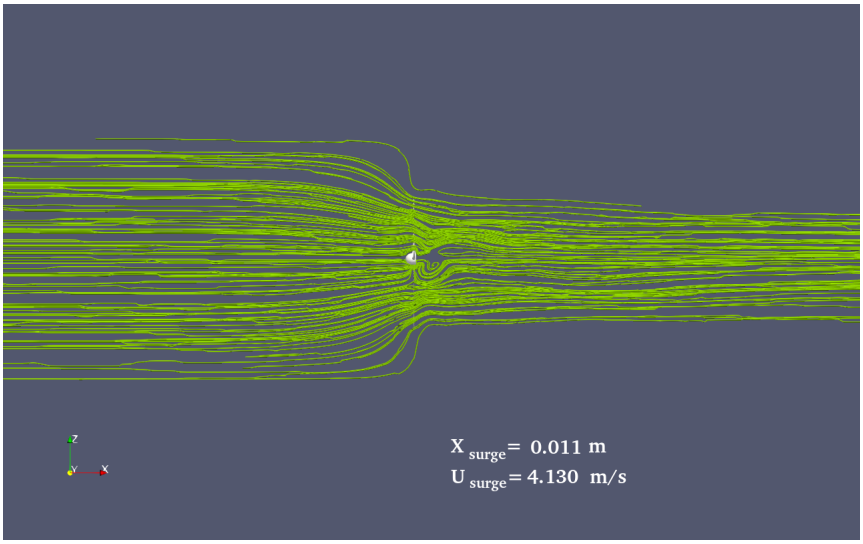


Figure 6.23: Streamlines of the velocity field relative to the rotor for the time instance $\frac{t}{T} = 0.5$

6.2.1.2 Scenes for the investigation of vortex ring state

In Figures 6.24 and 6.25, contours of y and z components of vorticity are shown, respectively. It is observed that vortical structures are formed as expected mostly on the tips and root of the rotor and are advected downstream. Close to $\frac{t}{T} = 0.5$, the relative velocity experienced by the rotor is expected to instantly become 0 and the rotor starts moving backwards hitting the previously formed vortices. All the time instances, before, after, or exactly at $\frac{t}{T} = 0.5$ were visualized in the form of vorticity contours. No large vortex ring seems to be created. To provide further information about the 3D vortical structures, isosurfaces of the q-criterion values are created. More than 30 different values of the q-criterion were tested and their respective isosurfaces were visualized. In all the scenes and animations produced, no vortex ring seems to be created. The isosurface of $q - criterion = 30$ is presented in Figure 6.26 since it provides the clearest view of the rotor's vortical structures.

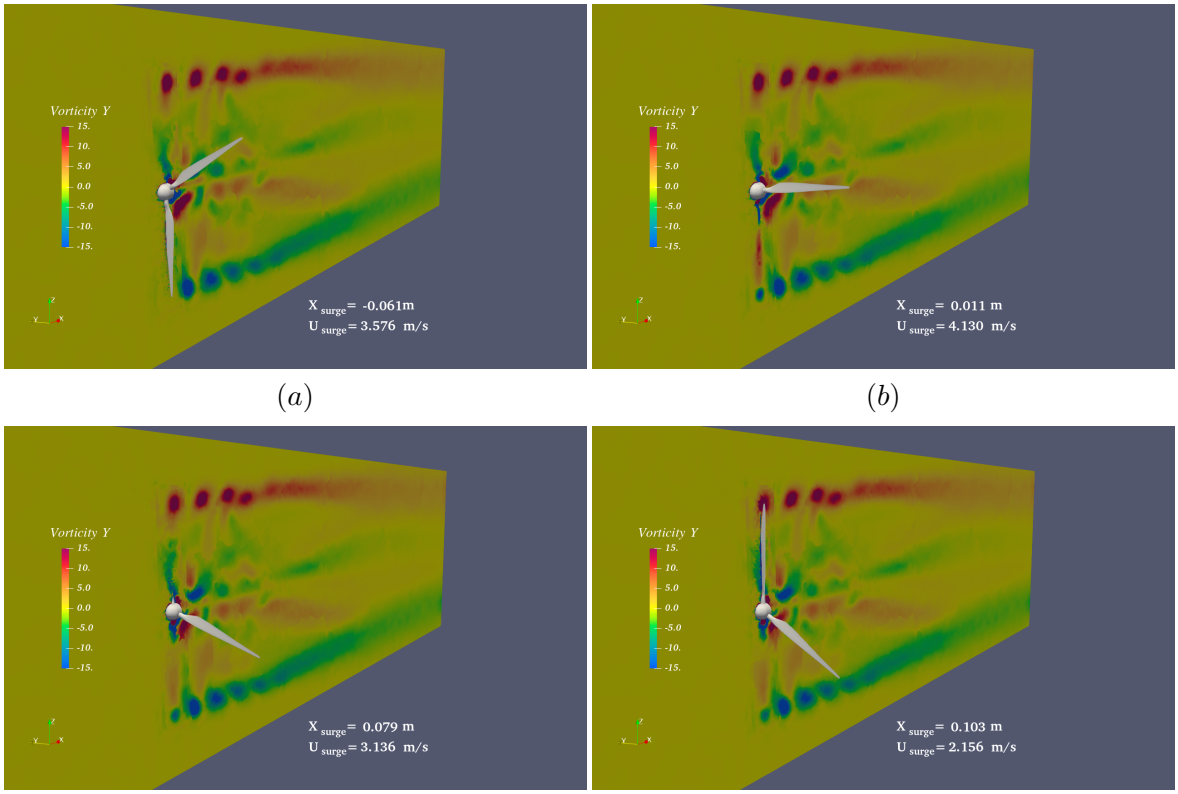


Figure 6.24: Contour of the y-component of vorticity on y-plane, i.e. on the side-view of the rotor. Four time instances are shown:(a): $\frac{t}{T} = 0.41$ (b): $\frac{t}{T} = 0.5$ (c): $\frac{t}{T} = 0.61$ (d): $\frac{t}{T} = 0.66$.

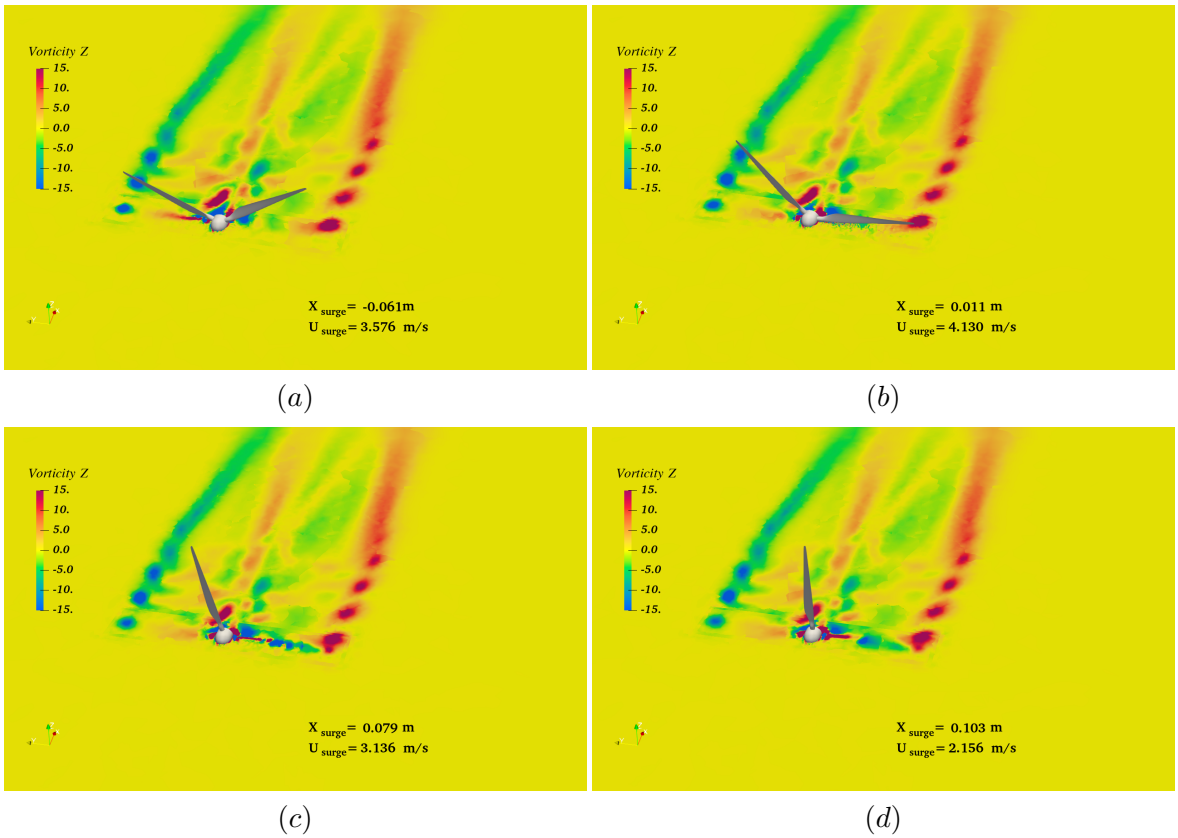
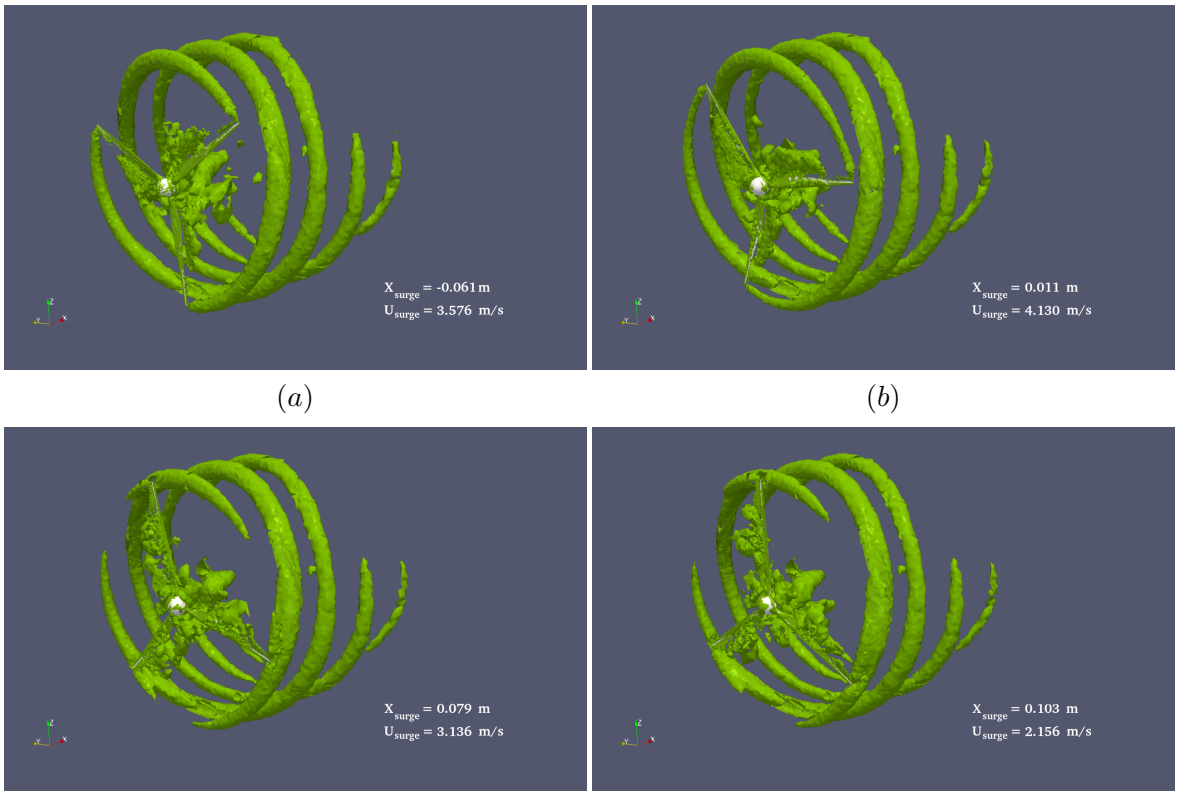


Figure 6.25: Contour of the z-component of vorticity on z-plane, i.e. on the top-view of the rotor. Four time instances are shown: (a): $\frac{t}{T} = 0.41$ (b): $\frac{t}{T} = 0.5$ (c): $\frac{t}{T} = 0.61$ (d): $\frac{t}{T} = 0.66$.



(a) (b)

(c) (d)

Figure 6.26: Isosurface of $q - criterion = 30$. Four time instances are shown: (a): $\frac{t}{T} = 0.41$ (b): $\frac{t}{T} = 0.5$ (c): $\frac{t}{T} = 0.61$ (d): $\frac{t}{T} = 0.66$.

Chapter 7

Conclusions and Recommendations

In the present thesis a blade-resolved CFD model was employed for the analysis of the unsteady aerodynamics of a surging rotor. A literature review was performed in chapter 2 revealing that there are only a few researchers that employed a high-fidelity blade-resolved CFD model for their investigation and that very limited efforts were done in answering whether propeller or vortex ring states occur during the operation of a FOWT. Thus, the focus of the present thesis was decided based on these two points. A high-fidelity blade-resolved CFD model was built and it was employed to assess the performance of momentum theory and to provide clarity on whether propeller/vortex ring states occur. A scaled turbine identical to the one examined in the UNAFLOW [37] project was chosen and two test cases were examined. One with surge parameters identical to the high-fidelity experiment of UNAFLOW to validate the CFD model and one with surge parameters leading to $V_{max,red} > 1$ that would cause large C_T oscillations with even negative C_T values. The methodology that was followed to construct the CFD model as well as the Ferreira-Micallef model which was chosen to compute induction based on CFD data were discussed in chapter 4. Next, in chapter 5 the CFD model was validated with the experimental results of UNAFLOW and, finally in chapter 6, the results of the two test cases were presented. Comparisons between the induction factors computed by Ferreira-Micallef model from CFD data and by momentum theory as well as 3D visualizations of the CFD flow field were employed to answer the two research questions of the present thesis.

7.1 Conclusions

In the beginning of the present thesis, three research objectives were set. After reviewing the results of the CFD simulations, the following comments can be made on whether the present thesis reached these objectives:

1. *Build a high-fidelity blade-resolved CFD model in OpenFoam and validate it with high-*

fidelity experimental data.

As discussed in the introduction of this thesis, the availability of high-fidelity CFD codes and wind tunnel measurements is of paramount importance in order to validate the codes of lower fidelity like BEM but also to gain further insight on complex aerodynamic phenomena like propeller or vortex ring state. After reviewing the literature, it was found that very few researchers employed a high-fidelity blade-resolved CFD model and even fewer validated their CFD model with high-fidelity experimental data. The fact that high-fidelity blade-resolved CFD models for FOWT are so rare proves how demanding it is to build such a model.

In the present thesis, a blade-resolved CFD model was constructed in OpenFoam. In section 4.1 the simulation set-up of the CFD model was discussed. Spatial and temporal discretization of the NS equations were kept to second-order accuracy and the widely-used SST model was chosen to resolve turbulence. Since a rotor-only configuration (no tower or nacelle) was chosen to limit the complexity of the model, computationally expensive mesh motion techniques like the overset(chimera) or morphing techniques were avoided. As discussed in section 4.1.6, a sliding mesh technique was employed to resolve the rotation of the blades while a whole-domain (rigid body) mesh motion technique was employed to resolve the translational surging motion. A very high-quality hybrid hexahedra/tetrahedra mesh with $y^+ \leq 1$ was constructed in the commercial software Pointwise and a grid independence study was carried out in section 4.4 making sure that the wind turbine load calculations are independent of the mesh density.

In section 5.1, the CFD model was first validated for the fixed-rotor (non-surging) case. The model results were compared to the high-fidelity experimental and CFD data of UNAFLOW. Even though the CFD model of the present thesis is a relatively simplified model, it was observed that the it performed exceptionally well since it yielded very similar results to UNAFLOW. When comparing to the high-fidelity experiment of UNAFLOW, the model yielded an error of 3.75% for the thrust and 0.69% for the power. While when comparing with the highly sophisticated CFD model of UNAFLOW, the model yielded exceptionally low errors of 1.88% for the thrust and 0.09% for the power. After validating for the fixed-rotor case, the CFD model was also validated in section 5.2 for the surging-rotor case. When comparing the thrust oscilation predicted by the thesis CFD model to the one predicted by the UNAFLOW experiment it was observed that the CFD model follows the shape of the oscillation exceptionally well but it under-predicts the oscillation's maximals and minimals. Even though this under-prediction was shown to be of the order of $10 - 15\%$, it was also exhibited that the thesis CFD model had the same level of error with the UNAFLOW CFD model in the prediction of the thrust oscillation magnitude. Thus, the thesis CFD model performs surprisingly well when taking into account that it aimed for a low computational cost by using only second-order spatial discretization schemes and omitting the nacelle and the tower from the analysis.

2. *Answer whether momentum theory is able to capture the induction field of a FOWT under surge motion.*

   To answer this question, a way to compute the induction field from the CFD data is first needed. In the present thesis, Ferreira-Micallef induction computation model was employed to compute the induction of the simulated rotor. Induction factor calculation

using CFD data as input is not a trivial process. In fact, there is much ambiguity on what is the right process to produce a reliable induction field out of the CFD velocity field. After reviewing all the different induction computation methods Rahimi et al. [50] came into some conclusions. One of the most important of them was that even though 3D high-fidelity CFD data are employed in all of these methods, simplified 2D theory is employed to extract the induction factors. Thus, when 3D flow is present on the blades, the methods - including Ferreira-Micallef model - do not have a mechanism to account for chordwise bound circulation and thus are unreliable. For all other cases where 3D flow is not present, all induction computation methods seemed to deliver similar results hence making them reliable. Therefore, the induction field computed by Ferreira-Micallef model should not be considered as the basis from which the error of momentum theory can be computed but rather as induction field data to which momentum theory data can be compared. As discussed in section 2.2, momentum theory is also a 2D model and thus it is also not valid when 3D flow is present. Therefore, a comparison of the induction fields predicted by the thesis CFD model using Ferreira-Micallef model and momentum theory only makes sense when 3D flow is not present on the blades.

Nevertheless, in sections 6.1 and 6.2, momentum theory induction field was compared to the induction field computed by Ferreira-Micallef model for all root (5%), mid-span (50%) and tip (95%) regions of the blade. The only case where 3D radial flow was not present, and the comparison of the induction fields was meaningful, was for the mid-span region of the blade and for the mild surge conditions of case #1. In this comparison, the two induction fields had a similar shape and the percentile difference of momentum theory to Ferreira-Micallef model ranged between 2% and 10% with an average of only 4.5%. Thus, momentum theory seems to deliver very similar results to the ones predicted by Ferreira-Micallef model when 3D radial flow is not present. This observation however does not exactly answer the question that was set as a research objective but merely gives an indication that momentum theory can give reasonable results for a FOWT when 3D radial flow is not present. The only way to truly assess the performance of momentum theory in the induction field prediction of FOWT is to compare its results to an induction computation model that makes use of 3D CFD velocity data but also has a mechanism to account for 3D flow.

3. *Provide clarity on whether propeller and vortex ring (VRS) states take place during the operation of the FOWT under high surge frequencies/amplitudes.*

   For this research objective, the high surge frequencies/amplitudes of test case #2 were examined. The surge conditions of test case #2 lead to $V_{max,red} > 1$ and thus at time instance $\frac{t}{T} = 0.5$ during the surge period V_{surge} becomes larger than V_{inf}, as shown in Figure 6.10. For this time instance, negative values of C_T are also acquired, as shown in Figure 6.11, which based on some authors indicate the existence of propeller state. Furthermore, at the same time instance $\left(\frac{t}{T} = 0.5\right)$ the rotor starts moving backwards, hits on its own wake, and surge velocity reaches its maximum value. In this instant the effective wind speed experienced by the rotor may drop significantly, induction may instantly become equal to 1 and a vortex ring may be created.

   Three-dimensional visualisations of the CFD flow field were made with Paraview to draw conclusions on whether propeller or vortex ring states occur. In the streamlines scenes of

Figures 6.20 and 6.21, it was shown that the streamtube is always expanding, and thus it is in windmill state, even though C_T acquires negative values for $\frac{t}{T} = 0.5$. At $\frac{t}{T} = 0.5$ the relative velocity experienced by the rotor becomes negative the flow is travelling from the back to the front of the rotor. Thus, someone could argue that in Figures 6.20(b) and 6.21(b), since the flow is travelling from right to left, the streamtube is contracting and hence it is in propeller state. This logic, however, is erroneous since the streamlines of Figures 6.20 and 6.21 are produced with the inertial reference frame velocity and not with the relative reference frame velocity. In Figure 6.23, the streamlines produced based on the relative reference frame velocity were shown. It was observed, once again, that the streamtube is expanding but this time from right to left since the velocity is negative.

In Figures 6.24-6.26, contours of vorticity as well as isosurfaces of the q-criterion were presented to provide clarity on whether vortex ring state occurs. In all the scenes that were produced there was no large vortex ring visible in the wake, thus there is no indication that vortex ring state occurs for a FOWT surging under these harsh surge conditions.

7.2 Recommendations

This thesis aimed in the construction of a high-fidelity blade-resolved CFD model for FOWT and made use of this model to answer some of the most ambiguous topics regarding the aerodynamics of FOWT. Improvements could be made on the CFD model itself and several new research questions could be set using the present or similar CFD model as the means to answer them.

In terms of improving the CFD model of the present thesis, two things could be mainly added:

1. Inclusion of tower and nacelle into the model could increase the accuracy of the solution depending on the application solved but at the same time they would increase the computational cost.
2. Higher order spatial and temporal discretizations could also increase the accuracy of the solution with the expense of adding to the computational cost.

Several new research topics could be potentially investigated using the high-fidelity CFD model of the present thesis as the starting point:

1. The current thesis focused on the effect of surge motion on the aerodynamics of FOWT. However, as mentioned in section 1.3.2, pitch motion is another important motion that could affect the FOWT aerodynamics. For example, investigating whether propeller and /or vortex ring states severe occur during extreme pitch motions could be a promising topic.

2. During the surge motion of a FOWT, flapping motion of the blade tips could be caused. Investigating to what degree such flapping motions could affect the aerodynamics of FOWT could be a potential research objective.

3. Investigating more extreme surge conditions that would lead to larger $V_{max,red}$ could also be interesting to see whether any of the present conclusions are challenged. However, someone could argue that more extreme surge conditions could be unrealistic for the operation of FOWT.

4. Developing an induction computation method that takes as input the CFD velocity data of the present thesis but also accounts for 3D radial flow could be a very promising research topic. Using the induction field of such a model as a basis, the error of low-fidelty methods like momentum theory could be computed for the 3D flows present during the operation of FOWT.

Bibliography

[1] Unfccc: Paris agreement. december 2015.

[2] *IEA (2020), Renewables Information: Overview, IEA, Paris.* URL https://www.iea.org/reports/renewables-information-overview.

[3] *GWEC, Global Wind Report 2018. April 2019.* URL https://gwec.net/global-wind-report-2018/.

[4] *IEA (2019), Offshore Wind Outlook 2019, IEA, Paris.* URL https://www.iea.org/reports/offshore-wind-outlook-2019.

[5] JG Schepers, T Lutz, K Boorsma, S Gomez-Iradi, I Herraez, L Oggiano, H Rahimi, P Schaffarczyk, G Pirrung, and H Aa Madsen. Final report of iea wind task 29 mexnext (phase 3). 2018.

[6] T. T. Tran and D. H. Kim. The aerodynamic interference effects of a floating offshore wind turbine experiencing platform pitching and yawing motions. *Journal of Mechanical Science and Technology*, 29(2):549–561, 2015. ISSN 1738-494x.

[7] Finn Gunnar Nielsen, Tor David Hanson, and Bjorn Skaare. Integrated dynamic analysis of floating offshore wind turbines. In *International Conference on Offshore Mechanics and Arctic Engineering*, volume 47462, pages 671–679. ISBN 0791847462.

[8] Thomas Sebastian and MA Lackner. Characterization of the unsteady aerodynamics of offshore floating wind turbines. *Wind Energy*, 16(3):339–352, 2013. ISSN 1095-4244.

[9] Jason Mark Jonkman. Dynamics modeling and loads analysis of an offshore floating wind turbine. Report, National Renewable Energy Lab.(NREL), Golden, CO (United States), 2007.

[10] J. M. Jonkman and D. Matha. Dynamics of offshore floating wind turbines-analysis of three concepts. *Wind Energy*, 14(4):557–569, 2011. ISSN 1095-4244.

[11] D. Matha, M. Schlipf, A. Cordle, R. Pereira, and J. Jonkman. Challenges in simulation of aerodynamics, hydrodynamics, and mooring-line dynamics of floating offshore wind turbines. In *21st International Offshore and Polar Engineering Conference, ISOPE-2011*, pages 421–428. ISBN 10986189 (ISSN); 9781880653968 (ISBN).

[12] Amy N Robertson and Jason M Jonkman. Loads analysis of several offshore floating wind turbine concepts. In *The Twenty-first International Offshore and Polar Engineering Conference*. International Society of Offshore and Polar Engineers.

[13] Jan Gerhard Schepers. Engineering models in wind energy aerodynamics: Development, implementation and analysis using dedicated aerodynamic measurements. 2012.

[14] O de Vries. Fluid dynamic aspects of wind energy conversion. Report, Advisory Group for Aerospace Research and Development NEUILLY-SUR-SEINE (France), 1979.

[15] J. N. Sorensen and G. A. M. van Kuik. General momentum theory for wind turbines at low tip speed ratios. *Wind Energy*, 14(7):821–839, 2011. ISSN 1095-4244.

[16] MB Anderson, DJ Milborrow, and JN Ross. Performance and wake measurements on a 3 m diameter horizontal axis wind turbine. comparison of theory, wind tunnel and field test data. In *Int. Symp. Wind Energy Syst., Proc.;(United Kingdom)*, volume 2.

[17] H. Glauert. Airplane propellers, division l in aerodynamic theory. volume IV. Springer:Berlin, 1935.

[18] P. K. Chaviaropoulos and M. O. L. Hansen. Investigating three-dimensional and rotational effects on wind turbine blades by means of a quasi-3d navier-stokes solver. *Journal of Fluids Engineering-Transactions of the Asme*, 122(2):330–336, 2000. ISSN 0098-2202.

[19] W. Z. Shen, R. Mikkelsen, J. N. Sorensen, and C. Bak. Tip loss corrections for wind research turbine computations. *Wind Energy*, 8(4):457–475, 2005. ISSN 1095-4244.

[20] S. Mancini. *An experimental, analytical and numerical study of FOWT's unsteady aerodynamics. MS thesis, Politecnico di Milano, Milan, Italy, 2020.*

[21] Herman Snel. Application of a modified theodorsen model to the estimation of aerodynamic forces and aeroelastic stability. In *European Wind Energy Conference*.

[22] Mohamed Hammam. Analytical unsteady aerodynamic models for horizontal axis wind turbines. 2016.

[23] Anurag Rajan and Fernando L. Ponta. A novel correlation model for horizontal axis wind turbines operating at high-interference flow regimes. *Energies*, 12(6), 2019. ISSN 1996-1073. doi: 10.3390/en12061148. URL https://www.mdpi.com/1996-1073/12/6/1148.

[24] J. N. Sørensen, W. Z. Shen, and X. Munduate. Analysis of wake states by a full-field actuator disc model. *Wind Energy*, 1(2):73–88, 1998. doi: https://doi.org/10.1002/(SICI)1099-1824(199812)1:2⟨73::AID-WE12⟩3.0.CO;2-L. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/%28SICI%291099-1824%28199812%291%3A2%3C73%3A%3AAID-WE12%3E3.0.CO%3B2-L.

[25] D. M. Eggleston and F. Stoddard. *Wind turbine engineering design.* Van Nostrand Reinhold Company, 1987.

[26] Ryan Kyle, Yeaw Chu Lee, and Wolf-Gerrit Früh. Propeller and vortex ring state for floating offshore wind turbines during surge. *Renewable Energy*, 155:645–657, 2020. ISSN 0960-1481. doi: https://doi.org/10.1016/j.renene.2020.03.105.

[27] Jing Dong and Axelle Viré. Comparative analysis of different criteria for the prediction of vortex ring state of floating offshore wind turbines. *Renewable Energy*, 163:882–909, 2021. ISSN 0960-1481. doi: https://doi.org/10.1016/j.renene.2020.08.027. URL https://www.sciencedirect.com/science/article/pii/S0960148120312647.

[28] C. Ferreira, W. Yu, A. Sala, and A. Vire. Dynamic inflow model for a floating horizontal axis wind turbine in surge motion. *Wind Energy Science Discussions 2021*, pages 1–22, 2021. doi: 10.5194/wes-2021-34.

[29] J. B. de Vaal, M. O. L. Hansen, and T. Moan. Effect of wind turbine surge motion on rotor thrust and induced velocity. *Wind Energy*, 17(1):105–121, 2014. ISSN 1095-4244.

[30] S Øye. A simple vortex model of a turbine rotor. 1990.

[31] D. Micallef and T. Sant. Loading effects on floating offshore horizontal axis wind turbines in surge motion. *Renewable Energy*, 83:737–748, 2015. ISSN 0960-1481.

[32] R. Farrugia, T. Sant, and D. Micallef. A study on the aerodynamics of a floating wind turbine rotor. *Renewable Energy*, 86:770–784, 2016. ISSN 0960-1481.

[33] T. T. Tran and D. H. Kim. A cfd study into the influence of unsteady aerodynamic interference on wind turbine surge motion. *Renewable Energy*, 90:204–228, 2016. ISSN 0960-1481.

[34] L. Bernini, I. Bayati, D. M. Boldrin, M. Cormier, M. Caboni, and R. F Mikkelsen. Unsteady aerodynamics for floating wind (unaflow). *Report, Politecnico di Milano, Milan, Italy*, 2018.

[35] SIMONE MANCINI. An experimental, analytical and numerical study of fowt's unsteady aerodynamics. 2020.

[36] M. Cormier, M. Caboni, T. Lutz, K. Boorsma, and E. Krämer. Numerical analysis of unsteady aerodynamics of floating offshore wind turbines. In C. E. D. Riboldi, A. Croce, L. Sartori, and S. Cacciola, editors, *7th Science of Making Torque from Wind, TORQUE 2018*, volume 1037. Institute of Physics Publishing. ISBN 17426588 (ISSN). doi: 10.1088/1742-6596/1037/7/072048.

[37] Simone Mancini, Koen Boorsma, Marco Caboni, Marion Cormier, Thorsten Lutz, Paolo Schito, and Alberto Zasso. Characterization of the unsteady aerodynamic response of a floating offshore wind turbine to surge motion. *Wind Energy Science*, 5(4):1713–1730, 2020. ISSN 2366-7443.

[38] Z. W. Chen, X. D. Wang, Y. Z. Guo, and S. Kang. Numerical analysis of unsteady aerodynamic performance of floating offshore wind turbine under platform surge and pitch motions. *Renewable Energy*, 163:1849–1870, 2021. ISSN 0960-1481.

[39] J. Wu, J. H. Ding, Y. P. He, and Y. S. Zhao. Study on unsteady aerodynamic performance of floating offshore wind turbine by cfd method. In *25th International Ocean and Polar Engineering Conference, ISOPE 2015*, volume 2015-January, pages 554–560. International Society of Offshore and Polar Engineers. ISBN 10986189 (ISSN); 9781880653890 (ISBN).

[40] Y. Fang, G. Li, L. Duan, Z. L. Han, and Y. S. Zhao. Effect of surge motion on rotor aerodynamics and wake characteristics of a floating horizontal-axis wind turbine. *Energy*, 218, 2021. ISSN 0360-5442.

[41] T. Sebastian and M. Lackner. Analysis of the induction and wake evolution of an offshore floating wind turbine. *Energies*, 5(4):968–1000, 2012. ISSN 1996-1073.

[42] Ilmas Bayati, Marco Belloli, Luca Bernini, and Alberto Zasso. Aerodynamic design methodology for wind tunnel tests of wind turbine rotors. *Journal of Wind Engineering and Industrial Aerodynamics*, 167:217–227, 2017. ISSN 0167-6105. doi: https://doi.org/10.1016/j.jweia.2017.05.004.

[43] M Sayed, Th Lutz, and E Kramer. Aerodynamic investigation of flow over a multi-megawatt slender bladed horizontal-axis wind turbine. *Renewable Energies Offshore*, pages 773–780, 2015.

[44] Florian R Menter. Two-equation eddy-viscosity turbulence models for engineering applications. *AIAA journal*, 32(8):1598–1605, 1994. ISSN 0001-1452.

[45] *OpenFoam v2012 documentation*. URL https://www.openfoam.com/documentation/guides/latest/doc/openfoam-guide-boundary-conditions.html.

[46] *ANSYS Fluent documentation*. URL https://www.afs.enea.it/project/neptunius/docs/fluent/html/ug/node238.htm.

[47] I. Demirdzic, Z. Lilek, and M. Peric. A collocated finite-volume method for predicting flows at all speeds. *International Journal for Numerical Methods in Fluids*, 16(12):1029–1050, 1993. ISSN 0271-2091.

[48] I. Demirdzic and S. Muzaferija. Numerical-method for coupled fluid-flow, heat-transfer and stress-analysis using unstructured moving meshes with cells of arbitrary topology. *Computer Methods in Applied Mechanics and Engineering*, 125(1-4):235–255, 1995. ISSN 0045-7825.

[49] ANSYS. Ansys fluent 12.0 theory guide - 3.2 sliding mesh theory. 2009. URL https://www.afs.enea.it/project/neptunius/docs/fluent/html/th/node37.htm.

[50] H. Rahimi, J.G. Schepers, W.Z. Shen, N. Ramos García, M.S. Schneider, D. Micallef, C.J. Simao Ferreira, E. Jost, L. Klein, and I. Herráez. Evaluation of different methods for determining the angle of attack on wind turbine blades with cfd results under axial inflow conditions. *Renewable Energy*, 125:866–876, 2018. ISSN 0960-1481. doi: https://doi.org/10.1016/j.renene.2018.03.018. URL https://www.sciencedirect.com/science/article/pii/S0960148118303239.

[51] C Lindenburg. Investigation into rotor blade aerodynamics. *ECN, Peten*, 2003.

[52] Wen Zhong Shen, Martin O. L. Hansen, and Jens Nørkær Sørensen. Determination of angle of attack (aoa) for rotating blades. In Joachim Peinke, Peter Schaumann, and Stephan Barth, editors, *Wind Energy*, pages 205–209, Berlin, Heidelberg, 2007. Springer Berlin Heidelberg. ISBN 978-3-540-33866-6.

[53] Martin O. L. Hansen, Niels N. Sørensen, Jens Nørkær Sørensen, and Jess A. Michelsen. Extraction of lift, drag and angle of attack from computed 3-d viscous flow around a rotating blade. In R. Watson, editor, *Proceedings of the European wind energy conference*, pages 499–502. Irish Wind Energy Association, 1998. ISBN 0-9533922-0-1. 1997 European Wind Energy Conference , EWEC '97 ; Conference date: 06-10-1997 Through 09-10-1997.

[54] Wen Zhong Shen, Martin O. L. Hansen, and Jens Nørkær Sørensen. Determination of the angle of attack on rotor blades. *Wind Energy*, 12(1):91–98, 2009. doi: https://doi.org/10.1002/we.277. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/we.277.

Appendix A

Induction Computation Script

The python script that was utilized in the present thesis to compute the induction factor inside Paraview can be seen below:

```
#### import the simple module from the paraview
from paraview.simple import *
import math
import matplotlib.colors as colors
from matplotlib import path
from scipy.optimize import least_squares
from scipy.spatial import distance
from scipy import interpolate
import numpy as np
import matplotlib.pyplot as plt
from paraview import numpy_support as ns
#### disable automatic camera reset on 'Show'
paraview.simple._DisableFirstRenderCameraReset()

# find source
foamfoam = FindSource('foam.foam')

#define angular rotation speed of the rotor (rad/s)
Omega=25.237461
fs=2 #Hz
As=0.008 #m

# create a new 'Transform'
transform1 = Transform(registrationName='Transform1', Input=foamfoam)
```

```
# toggle 3D widget visibility (only when running from the GUI)
Hide3DWidgets(proxy=transform1.Transform)

# get animation scene
animationScene1 = GetAnimationScene()
#define real_time vector
real_time=animationScene1.TimeKeeper.TimestepValues

# Properties modified on transform1.Transform
transform1.Transform.Rotate = [-real_time[0]*Omega*(180/np.pi), 0.0, 0.0]

#UpdatePipeline(time=real_time[0], proxy=transform1)

# create a new 'Cell Data to Point Data'
cellDatatoPointData1 = CellDatatoPointData(registrationName='CellDatatoPointData1',
Input=transform1)

# Properties modified on cellDatatoPointData1
cellDatatoPointData1.PassCellData = 1

#UpdatePipeline(time=real_time[0], proxy=cellDatatoPointData1)

# create a new 'Slice'
slice1 = Slice(registrationName='Slice1', Input=cellDatatoPointData1)

# toggle 3D widget visibility (only when running from the GUI)
Hide3DWidgets(proxy=slice1.SliceType)

# Properties modified on slice1
slice1.Triangulatetheslice = 0

# Properties modified on slice1.SliceType
slice1.SliceType.Origin = [0.0, 0.0, 0.1]
slice1.SliceType.Normal = [0.0, 0.0, 1.0]

#UpdatePipeline(time=real_time[0], proxy=slice1)

# create a new 'Point Data to Cell Data'
pointDatatoCellData1 = PointDatatoCellData(registrationName='PointDatatoCellData1',
Input=slice1)

# Properties modified on pointDatatoCellData1
pointDatatoCellData1.PassPointData = 1

# create a new 'Transform' TRANSFORMING TO GET THE SAME X-Y-Z ORIENTATION AS
THE ONE SHOWN IN CARLOS PAPER
transform_paper_orientation = Transform(registrationName=
```

```
'transform_paper_orientation', Input=pointDatatoCellData1)
# toggle 3D widget visibility (only when running from the GUI)
#Hide3DWidgets(proxy=transform_paper_orientation.Transform)
# Properties modified on transform1.Transform
transform_paper_orientation.Transform.Rotate = [180.0, 0.0, 90.0]

#### Extract internal Mesh from the transformed slice ##########
extract_internal = ExtractBlock(registrationName='extract_internal',
Input=transform_paper_orientation)
# Properties modified on extractBlock1
extract_internal.Selectors = ['/Root/internalMesh']

############ Extract airfoil coords #############
extract_airfoil = ExtractBlock(registrationName='extract_airfoil',
Input=transform_paper_orientation)
# Properties modified on extractBlock1
extract_airfoil.Selectors = ['/Root/boundary']
cellCenters = CellCenters(registrationName='CellCenters',
Input=extract_airfoil)
# Properties modified on cellCenters1
cellCenters.VertexCells = 1
mergedairfoilblockresults = MergeBlocks(registrationName=
'mergedairfoilblockresults', Input=cellCenters)
airfoil_coords = Calculator(registrationName='airfoil_coords',
Input=mergedairfoilblockresults)
airfoil_coords.ResultArrayName = 'airfoil_coords'
airfoil_coords.Function = 'coords'



# create a new 'Cell Centers'
cellCenters1 = CellCenters(registrationName='CellCenters1',
Input=extract_internal)
# Properties modified on cellCenters1
cellCenters1.VertexCells = 1


## MULTIBLOCK RESULTS NEEDED##
mergedBlockresults = MergeBlocks(registrationName='mergedBlockresults',
Input=cellCenters1)

###### GET CELL CENTRE COORDS OF ALL THE INTERNAL CELLS#####
coords = Calculator(registrationName='coords', Input=mergedBlockresults)
coords.ResultArrayName = 'coords'
coords.Function = 'coords'

####### create a new 'Cell Size' filter to get the 2D cell Area ######
```

```
cellSize1 = CellSize(registrationName='CellSize1', Input=extract_internal)
# Properties modified on cellSize1
cellSize1.ComputeVertexCount = 0
cellSize1.ComputeLength = 0
cellSize1.ComputeVolume = 0
## MERGED MULTIBLOCK CELL AREAS NEEDED TO EXTRACT A NUMPY ARRAY##
mergedBlockarea = MergeBlocks(registrationName='mergedBlockarea',
Input=cellSize1)


# 2D Biot-Savart equation
def vortex2D(gamma,xvortex,yvortex,xtarget,ytarget,coreradius=.0000001):
    dx=xtarget-xvortex
    dy=ytarget-yvortex
    rsquared=dx**2+dy**2
    ind1= (np.sqrt(rsquared)<coreradius)
    if np.any(ind1):
        rsquared[ind1]=1
        #rsquared=1
    u= -gamma/2/np.pi*dy/rsquared
    v= gamma/2/np.pi*dx/rsquared
    if np.any(ind1):
        u[ind1]=0
        v[ind1]=0
    return u,v


def error_velocity2(x,uxrem, uyrem,uxvormat,uyvormat):
    vortices=x[2:]
    uxvor = np.zeros(np.shape(uxrem))
    uyvor = np.zeros(np.shape(uyrem))
    for i in range(np.shape(uxvormat)[0]):
        for j in range(np.shape(uxvormat)[1]):
            uxvor[i]=uxvor[i]+uxvormat[i,j]*vortices[j]
            uyvor[i]=uyvor[i]+uyvormat[i,j]*vortices[j]

    # calculate velocity induced by vortices over airfoil

    error = np.append(uxrem-uxvor-x[0],uyrem-uyvor-x[1])
    error2=np.reshape(error, (np.product(error.shape)))
    error2=error2**2
#     print(np.shape(error2))
    return error2

def rotate_matrix (x, y, angle, units="DEGREES"):
    """
    Rotates a point in the xy-plane counterclockwise through an angle about the origin
```

```
    https://en.wikipedia.org/wiki/Rotation_matrix
    :param x: x coordinate
    :param y: y coordinate
    :param x_shift: x-axis shift from origin (0, 0)
    :param y_shift: y-axis shift from origin (0, 0)
    :param angle: The rotation angle in degrees
    :param units: DEGREES (default) or RADIANS
    :return: Tuple of rotated x and y
    """

    # Convert degrees to radians
    if units == "DEGREES":
        angle = math.radians(angle)

    #print('x=',x)
    #print('y=',y)
    coord=np.array([x,y])
    #print('coord=',coord)
    #print('coord.shape=',coord.shape)
    ##coord=np.transpose(coord)
    ##print('new coord.shape=',coord.shape)

    R=np.array([[np.cos(angle),-np.sin(angle)],[np.sin(angle),np.cos(angle)]])
    #print('R shape=',R.shape)

    # Rotation matrix multiplication to get rotated x & y
    result=np.dot(R,coord)
    #print('result.shape=',result.shape)
    #print('result=',result)
    return result[0],result[1]

coord_list = [] ##only for computing the chord
#X_airfoil_list= []
#Y_airfoil_list= []
avg_induction=np.zeros((2,np.size(real_time)))
## Define loop to run in spanwise direction of the blade ###
percent_start=2
percent_end=98
percent_step=2
slice_loop=np.arange((percent_start/100)*1.09937+0.0869415,
(percent_end/100)*1.09937+0.0869415+(percent_step/100)*1.09937,(percent_step/100)*1.09937)

### Compute Surface areas of the radial positions ###
r_pos=np.arange(((percent_start-1)/100)*1.09937+0.0869415,
((percent_end+1)/100)*1.09937+0.0869415+(percent_step/100)*1.09937,(percent_step/100)*1.09
A=np.pi*(r_pos[1:]**2) - np.pi*(r_pos[0:-1]**2)
```

```
for time_counter,time_value in enumerate(real_time[7:63]):
    dy_surge=-As*np.sin(2*np.pi*fs*time_value)
    spanwise_induction=np.zeros((3,np.size(slice_loop)))
    transform1.Transform.Rotate = [-time_value*Omega*(180/np.pi), 0.0, 0.0]
    for position,slice_z in enumerate(slice_loop):
        slice1.SliceType.Origin = [0.0, 0.0, slice_z]
        coord_list.clear()
        UpdatePipeline(time=time_value, proxy=airfoil_coords)
        ######GET AIRFOIL COORDS ################
        intData = servermanager.Fetch(airfoil_coords)
        airfoil_coords_array=ns.vtk_to_numpy(intData.GetPointData().
        GetArray('airfoil_coords'))
        ####### GET INTERNAL MESH COORDS ############
        intData = servermanager.Fetch(coords)
        coords_array=ns.vtk_to_numpy(intData.GetPointData().
        GetArray('coords'))
        ######GET AREA OF ALL CELLS ################
        intData = servermanager.Fetch(mergedBlockarea)
        area_array=ns.vtk_to_numpy(intData.GetCellData().
        GetArray('Area'))
        ###### GET VELOCITY AT CELL CENTRES ################
        intData = servermanager.Fetch(mergedBlockresults)
        U_array=ns.vtk_to_numpy(intData.GetPointData().GetArray('U'))
        ###### GET VORTICITY AT CELL CENTRES ################
        intData = servermanager.Fetch(mergedBlockresults)
        #vorticity_array=ns.vtk_to_numpy(intData.GetPointData().
        GetArray('vorticity'))
        vorticity_array=np.zeros(np.shape(U_array))

        X=np.array(coords_array[:,0])
        Y=np.array(coords_array[:,1])
        X_airfoil=np.array(airfoil_coords_array[:,0])
        Y_airfoil=np.array(airfoil_coords_array[:,1])
        ux=np.array(U_array[:,0])
        uy=np.array(U_array[:,1])
        vortz=np.array(vorticity_array[:,2])
        area=np.array(area_array)

        ######## Set X_airfoil,Yairfoil as the coordinates of the
        first layer cell cell centers. ##################
        for i,val in enumerate(X_airfoil):
            coord_list.append((X_airfoil[i],Y_airfoil[i]))
            # find nearest
            iind=np.argmin(((X - X_airfoil[i])**2 + (Y - Y_airfoil[i])**2)**0.5)
            X_airfoil[i]=X[iind]
            Y_airfoil[i]=Y[iind]
```

```
###### Find airfoil chord ##################
dists = distance.cdist(coord_list, coord_list, 'euclidean')
chord = np.max(dists)
maxdx = np.max(X_airfoil) - np.min(X_airfoil)
maxdy = np.max(Y_airfoil) - np.min(Y_airfoil)


######### Define Locations of Control Points xcp,ycp ####

dtheta2=np.pi/20
theta2= np.arange(0,4*np.pi,dtheta2) # angle of the points of the circle
radiuscale=np.linspace(1, 1.4, num=np.size(theta2))
dist_from_airfoil_surf=0.5*chord
xcp=np.cos(theta2)*(dist_from_airfoil_surf+maxdx/2)*radiuscale+
np.max(X_airfoil)-maxdx/2
ycp=np.sin(theta2)*(dist_from_airfoil_surf+maxdy/2)*radiuscale

###Rotate cp coords based on the airfoil rotation ####
if maxdx/chord < 1:
    angle_rotation=np.degrees(np.arccos(maxdx/chord))
    #print('angle_rot=',angle_rotation)
    xcp, ycp = rotate_matrix(xcp,ycp,angle_rotation, units="DEGREES")

### Correct the position of the control points based on the
### surge motion dy displacement
ycp=ycp+dy_surge

#### define velocity vectors for control points #####
uxcp=np.zeros(np.shape(xcp))
uycp=np.zeros(np.shape(xcp))
#### Define area vector for cells on control points####
area_cp=np.zeros(np.shape(xcp))
#Move the control points to mesh points and assign velocities
for i,val in enumerate(xcp):
    # find nearest
    iind=np.argmin(((X - xcp[i])**2 + (Y - ycp[i])**2)**0.5)
    xcp[i]=X[iind]
    ycp[i]=Y[iind]
    area_cp[i]=area[iind]
    uxcp[i]=ux[iind]
    uycp[i]=uy[iind]

# resulting difference
uxremcp=np.array(uxcp)
uyremcp=np.array(uycp)

#print('blade percent=',((slice_z-0.0869415)/1.09937)*100)
```

```
####### Add vorticity system over the airfoil surface ############
#Althoug much of the vrticity over the surface of the airfoil is
already accounted for when we calculated the vorticity,
#we will add an additional vorticity system to account for error.
Also, in the cse of moving airfoil, it can account for
#incorrect calcualtion fo the motion of the airfoil. Also, this
vorticy will replace the vortices closest to the surface of the
airfoil

#### add vortices over the surface
xvort = np.array(X_airfoil)
yvort = np.array(Y_airfoil)

### create induction matrices for vortex points
uxvormat=np.zeros([np.size(xcp),np.size(xvort)])
uyvormat=np.zeros([np.size(xcp),np.size(xvort)])
coreradius_cp = np.array(np.sqrt(area_cp/(np.pi)))*2

for i,val in enumerate(xcp):
    for j,val2 in enumerate(xvort):
        uxvormat[i,j],uyvormat[i,j]=vortex2D(1,xvort[j],yvort[j],
        xcp[i],ycp[i],coreradius_cp[i])

######## CALCULATE SOLUTION WITH LEAST SQUARES METHOD WHEN HAVING
EXTRA VORTICES ON AIRFOIL ######################

x0 = np.ones(2+np.size(xvort))
x0[0]=2
res_2 = least_squares(error_velocity2, x0,args=(uxremcp, uyremcp,
uxvormat, uyvormat))
Uvector2=res_2.x[0:2]
#print('The calculated Ux is: ', Uvector2[0], 'The calculated Uy
is: ', Uvector2[1])
#print('The calculated axial induction is: ', (4-Uvector2[1])/4)
spanwise_induction[0,position]= ((slice_z-0.0869415)/1.09937)*100
spanwise_induction[1,position]=(4-Uvector2[1])/4
spanwise_induction[2,position]= (Uvector2[0]/(Omega*slice_z))-1

if position == 35:
    ### Export last spanwise location airfoil and cp points ####
    np.savetxt("cp_coords_x.txt",xcp,newline=" ")
    np.savetxt("cp_coords_y.txt",ycp,newline=" ")
    np.savetxt("airfoil_coords_x.txt",X_airfoil,newline=" ")
    np.savetxt("airfoil_coords_y.txt",Y_airfoil,newline=" ")
```

```
    #filename = "/home/damon/spanwise_induction_with_UNAFLOW_surge_{:02f}.
    txt".format(time_value)
    #np.savetxt(filename,np.transpose(spanwise_induction))

    avg_induction[0,time_counter]=time_value
    avg_induction[1,time_counter]=sum(spanwise_induction[1,:]*A)/sum(A)


np.savetxt("avg_induction_with_UNAFLOW_surge_reconstructPar",
np.transpose(avg_induction))
print("RUN ENDED")
```