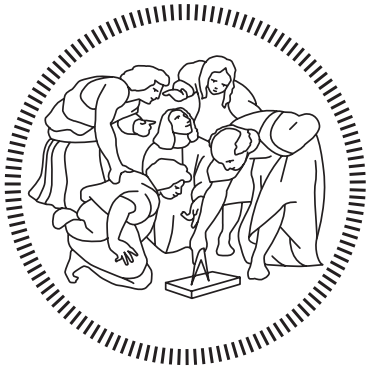
Politecnico di Milano

SCHOOL OF INDUSTRIAL AND INFORMATION ENGINEERING
Master of Science – Mechanical Engineering



Pitch and surge motion effects on floating wind turbine unsteady aerodynamics

Supervisor
Prof. Paolo SCHITO

Co-Supervisor
Prof. Alberto ZASSO
Ing. Claudia MUSCARI

Candidate
Filippo NEGRONI–920595

Academic Year 2021 – 2022

"Le vent se lève"

Ringraziamenti

Per questo lavoro di tesi vorrei ringraziare per primi Paolo Schito e Albrto Zasso, per avermi dato la possibilità di lavorare in un settore davvero interessante come quello dell'aerodinamica e delle turbine eoliche. Ringrazio inoltre Claudia per l'aiuto e assisenza durante questa tesi.
Ringrazio inoltre Simone Mancini per il grande lavoro svolto nella sua tesi, ricca riferimenti e per la condivisione di dati per le simulazioni. Un ringraizamneto speciale va sicuramente fatto alla mia famiglia allargata con cui sono rinchiuso in casa da più di una anno ormai ma che non mi stufa, quindi grazie a Virgilio, Giacinta, Lorella, Pier, Samantha, Fulvia, Umberto, Elisa e Maria, Un grazie poi a tutti i compagni con cui ho condiviso questi leggeri cinque anni di studi, quindi grazie a Andy, Placido, Gabri, Saso, Foghen, Niso, Luca Ono, Noraiz, Sergio, Amedeo e Matteo. Ringraziamento a parte dedicato Luca Gajoni, per avere condiviso questo ultimo anno di tesi e assieme. Un grazie poi soprattutto ai mei amici, perchè senza di loro sarei riuscito a laurearni con un paio d'anni d'anticipo ma non mi sarei divertito, quindi grazie ad Ale, Franci, *L*èris, **B**angsbo, Tia, Nicolò, Alice, Simone. Dulcis in fundo, messa apposta qui per la tua golosità, grazie a Virna

Sommario

Oggi la produzione di energia eolica è uno dei punti di riferimento per la transizione verde delle energie rinnovabili. In questo contesto negli ultimi anni sta guadagnando terreno il settore offshore, che cerca di sfruttare la logica dell'economia di scla per abbassare i costi di produzione dell'energia e di accoppiarla con altre tecnologie green, come la produzione di idrogeno dall'acqua tramite il surplus energetico delle turbine eoliche. Le dimensioni della macchina combinate con le asperità dei lughi in cui si trovano rendono le misurazioni sperimentale troppo complessa e costosa per un'analisi coerente dei dati sulle loro prestazioni e sul vento circostante. In questo contesto le simulazioni numeriche, abbinate ad un costante aumento delle risorse computazionali, hanno riscosso molta attenzione sia nell'ambito della ricerca che in quello industriale. Una simulazione CFD delle turbine vera e propria costa troppo in termini di tempo e risorse di calcolo. Sono stati quindi implementati codici numerici più semplici per simulare l'interazione equivalente turbina-flusso, e tra questi esistono i modelli Actuator Line, su cui POLIMI ha scritto e svilupptato un codice. Questo modello può essere visto come il compromesso tra una soluzione CFD completa e un codice iterativo come BEM. Si è ritenuto utile analizzare gli effetti non stazionari anche con codici numerici ibridi come questo. Per questo scopo un'implementazione nel codice già consentiva di imporre un moto di Surge prescritto al rotore della turbina e ne è stata eseguita una aggiuntivo in questo lavoro di tesi, vale a dire la movimentazione in pitch della piattaforma. Questo tipo di spostamento è considerato più rilevante rispetto agli altri, a causa delle condizioni di afflusso sbilanciate a cui è sottoposto il rotor; in un periodo di oscillazione di Surge il rotore è sempre perpendicolare alla direzione del flusso in ingresso. Per quantificare queste discrepanze tra i gradi di libertà della turbina flottante è stata infatti sviluppata un'analisi comparativa in questo lavoro di tesi, e i risultati di riferimento sono stati riferiti alla distribuzione dei carichi sulle pale e sulla generazione di scia a valle del rotore della turbina. I casi duali sono stati selezionati secondo un parametro sintetico chiamato velocità ridotta. I risultati della soluzione numerica mettono in luce la forte somiglianza in termini di carichi tra i casi di Surge e di Pitch a confronto, anche se l'evoluzione del profilo della scia sembra essere meno regolare dietro la macchina, causando un ulteriore squilibrio dei carichi sui potenziali allineati turbine in un parco eolico.
Parole chiave: turbine eoliche flottanti, codice actuator line, aerodinamica non stazionaria.

Abstract

Nowadays wind energy production is one of the benchmark for the renewable energy green transition. In this context during last years its floating offshore branch is gaining ground trying t exploiting the economy scale logic to lower the energy production costs and to couple it with other green technologies, like hydrogen production, from water by means of wind turbines energy surplus. The machine dimensions combined with the harshness of their locations make experimental set-up too complex and expensive for a consistent data analysis on their performance and the surrounding wind. In this context numerical simulations, coupled with a constant increase of the computational resources, gained a lot of attention both in research and industrial area. A fully-resolved CFD simulation of the turbines costs too much for in terms of computational time and resources. simpler numerical codes were hence implemented in order to simulate the equivalent turbine-flow force interaction, and among them there exist the Actuator Line (AL) models, on which POLIMI has based its in-house developed code. This model can be seen as the compromise between a complete CFD solution and an iterative code like BEM. It was found useful to analyse the unsteady aerodynamics effect also with numerical codes like this one. For this scope an implementation in the codes already allowed to impose a prescribed surge motion to the turbine rotor and a additional one was performed for this thesis work, namely he platform pitch motion. This kind of displacement is considered to be more relevant with respect to the other ones, due to the skewed inflow conditions to which is subjected; in a surge period of oscillation the rotor is always perpendicular to the incoming flow direction. In order to quantify these discrepancies among the floating turbine degrees of freedom a comparative analysis was indeed developed in this thesis work, and the benchmark outputs were referred to the loads distribution on the blades and on the wake generation downstream the turbine rotor. The dual cases were selected according to a synthetic parameter called reduced velocity.
The results of the numerical solution shed light on the strong similarity in terms of loads between the surge and pitch compared cases, even if the evolution of the wake profile seems to be less smooth behind the wake, causing an additional unbalance of loads on potential aligned turbines in a wind farm.

Keywords: Wind Turbines, Actuator Line, Floating Offshore Wind Turbines, Unsteady Aerodynamics

Contents

Ringraziamenti **v**

Sommario **vii**

Abstract **ix**

Contents **xiii**

List of Figures **xvi**

List of Tables **xvii**

1 Introduction **1**
1.1 Historical wind energy review 1
1.2 State of the art and forecasts 2
1.3 Offshore Wind Turbines 3
1.4 Research Status on FOWT 5
1.5 Unsteady Aerodynamics on Wind Turbines 6
1.5.1 Two-dimensional blade aerodynamics 6
1.5.2 Unsteady aerodynamics parameters 7
1.5.3 Pitch and Surge platform motion and limits of Linear Momentum Theory 9
1.6 Thesis Scope and Themes 12

2 Introduction to CFD **15**
2.1 CFD 15
2.2 The steps of a CFD simulation 16
2.2.1 Mathematical Model 16
2.2.2 Discretization method 16
2.2.3 Analysis of the numerical schemes 17
2.2.4 Solution method 18
2.3 Governing equations 18
2.3.1 Conservation laws 18
2.3.2 The equations of fluid mechanics 19
2.4 Turbulence modeling 20
2.4.1 Energy cascade and implications on CFD modeling 21
2.4.2 LES: Large Eddy Simulations 21
2.5 OpenFOAM 22
2.5.1 OpenFOAM implementation of NS equations 23

3 OC6 Program Phase III Definition **25**
3.1 Experimental campaign set-up . . . 25
3.1.1 Overview of the experimental set-up . . . 25
3.1.2 Scaled Turbine design . . . 26
3.1.3 Wake measurements system set-up . . . 28
3.2 Turbine blades update . . . 28
3.3 Load Cases definition . . . 29

4 Actuator Line Model **31**
4.1 Overview on AL . . . 31
4.2 Actuator Line Description . . . 32
4.3 Actuator Line Code Structure . . . 34
4.4 Code Implementation . . . 35

5 Simulations set-up **39**
5.1 Domain choice and discretization . . . 39
5.2 Turbulence Model Selection . . . 41
5.3 Constant . . . 41
5.4 Boundary Conditions . . . 42
5.5 System . . . 43
5.5.1 Time Step Definition . . . 44
5.6 Simulation Choice criteria . . . 45
5.7 Probes placement . . . 47
5.7.1 Azimuthal Averaging Technique (AAT) . . . 47
5.7.2 Wake Measurements . . . 49
5.8 Mesh Sensitivity Analysis . . . 49
5.8.1 Meshes lists and set-up . . . 49
5.8.2 Mesh sensitivity outputs and selection . . . 50

6 Comparison of simulations outputs **53**
6.1 Fixed turbines output . . . 53
6.1.1 Integral Performances . . . 53
6.1.2 Local Forces evaluation . . . 54
6.1.3 Steady Axial induction extraction . . . 55
6.1.4 Wake Analysis . . . 57
6.2 Surge load cases analysis . . . 59
6.2.1 Loads comparison with experimental data and fixed turbine cases 60
6.2.2 Wake analysis . . . 64
6.3 Pitch and Surge comparison analysis . . . 65
6.3.1 Loads comparison . . . 66
6.3.2 Wake Analysis . . . 69
6.4 AL pitch simulation results comparison with other codes . . . 70
6.4.1 Integral Thrust and Torque . . . 70
6.4.2 Local forces evaluation . . . 72
6.5 Conclusions . . . 75
6.6 Future Works . . . 76

A First Appendix **77**
A.1 List and Description of AL dictionaries 77
A.2 Pitch motion in AL code implementation 83

B Second Appendix **85**
B.1 OC6 Load Cases Table for Surge Motion 85

C Second Appendix **87**
C.1 Surge and Pitch simulations output 88
C.1.1 LC2.6 88
C.1.2 LC2.8 89
C.1.3 p2.6 91
C.1.4 p2.8 92

Acronyms **93**

Bibliography **98**

List of Figures

Figure 1.1 Global cumulative installed wind capacity from 2001 to 2020 . 2
Figure 1.2 Amount of new installations from 2016-2020 3
Figure 1.3 Expected wind capacity installations for the next five years . . 3
Figure 1.4 Typical floaters concepts for floating wind turbines 4
Figure 1.5 Two-dimensional velocity triangle of a wind turbine airfoil section 6
Figure 1.6 The 6 DoF of a Floating Wind Turbine 7
Figure 1.7 Inflow distribution for Surge motion 10
Figure 1.8 Inflow distribution for Pitch motion 10
Figure 1.9 The potential flow configuration with a FOWT 11
Figure 1.10 Axial induction (a) versus Thrust (T) 12

Figure 3.1 Wind profile transverse measurements output 26
Figure 3.2 DTU 10 MW Wind turbine scaled model schematic 27
Figure 3.3 Schematic of the wake measurements set-up 28
Figure 3.4 Table for Reynolds regime prediction for each radial station . . 29

Figure 4.1 Acutator Line-mesh intersection schematic 32
Figure 4.2 Regularization Kernel effect on neighbouring cells 33

Figure 5.1 Visualization of the background mesh plus absolute domain reference axes 40
Figure 5.2 Visualization of the mesh with the addition of wake refinement zone 40
Figure 5.3 Clip visualization to show the inner rotor cylinder refinement zone 41
Figure 5.4 Pitch and Surge studied cases according to $\Delta V/V_0$ parameter . 46
Figure 5.5 Schematic of the two consecutive blades counterbalance of downwash and upwash 48
Figure 5.6 Probes placements for wake measurements 49
Figure 5.7 Mesh sensitivity analysis outputs for Thrust and Torque . . . 51

Figure 6.1 Axial,Radial and Tangential force distribution for each rotor blade, in LC1.1 (top) and LC1.2 (bottom) 55
Figure 6.2 Lift and Drag distribution for each rotor blade, in LC1.1 (top) and LC1.2 (bottom) 56
Figure 6.3 Axial induction computed with AAT in LC1.1 (top) and LC1.2 (bottom) 56
Figure 6.4 Wake numerical and experimental profiles for RATED case . . 57
Figure 6.5 Wake numerical and experimental profiles for ABOVE case . . 58

Figure 6.6 Wake evolution in flow direction for LC1.1 (top) and LC1.2 (bottom) 59
Figure 6.7 Coefficient of variation span distribution for LC1.1 (top) and LC2.5 (middle) and LC2.7 (bottom) 61
Figure 6.8 Dynamic induction evaluation for LC2.5 (top) and LC2.7 (bottom) 63
Figure 6.9 Mean axial induction distribution along the blade comparison 63
Figure 6.10 Wake numerical and experimental comparison for LC2.5 (top) and LC2.7 (bottom) 64
Figure 6.11 Wake axial evolution for LC2.5 (top) and LC2.7 (bottom) . . 65
Figure 6.12 Phase shift distribution for the analyzed pitch and surge cases 66
Figure 6.13 Coefficient of variation distribution for pC2.5 (top) and pC2.7 (bottom) 67
Figure 6.14 AAT outputs for LC2.7 (red) and p2.7 (blue) load cases 68
Figure 6.15 Wake evolution comparison for equivalent pitch/surge cases measured with an horizontal transverse, both 2.5D (top) and 5D (bottom) downstream the rotor 69
Figure 6.16 Wake evolution comparison for equivalent pitch/surge cases measured with a vertical transverse, both 2.5D (top) and 5D (bottom) downstream the rotor 70
Figure 6.17 Phase shift comparison among the codes 71
Figure 6.18 Axial (left) and tangential (right) mean load distributions for p2.5 (top) and p2.7 (bottom) 72
Figure 6.19 Axial force Coefficient of variation distribution for p2.5 (top) and p2.7 (bottom) 73

Figure B.1 Defined table for surge oscillation load cases 85

Figure C.1 Average Loads on the three turbine blades (top) and Coefficient of Variation distribution (bottom) for LC2.6 88
Figure C.2 Velocity profile components evolution, measured by horizontal transverse, for LC2.6 89
Figure C.3 Coefficient of Variation of loads distribution for LC2.8 89
Figure C.4 Velocity profile components comparison with experimental data (top) and numerically estimated wake evolution, measured by horizontal transverse, for LC2.8 90
Figure C.5 Coefficient of Variation of loads distribution for p2.6 91
Figure C.6 Velocity profile evolution along axial direction measured both by horizontal (top) and vertical (bottom) transverse for p2.6 91
Figure C.7 Coefficient of Variation of loads distribution for p2.8 92
Figure C.8 Velocity profile evolution along axial direction measured both by horizontal (top) and vertical (bottom) transverse for p2.8 92

List of Tables

Table 1.1 *Reduced Frequency Analysis* 8

Table 3.1 *Scaled Turbine Model properties* 27
Table 3.2 *Steady Load Cases* 30

Table 5.1 *turbineProperties* set-up for fixed turbine cases 42
Table 5.2 *Boundary Conditions set-up* 43
Table 5.3 Surge Load Cases definition 46
Table 5.4 Pitch Load Cases definition 46
Table 5.5 Meshes main characteristics 50

Table 6.1 Integral performances of fixed turbine cases 53
Table 6.2 Thrust and Torque average values and oscillation amplitude . . 60
Table 6.3 Thrust and Torque average values and oscillation amplitude for equivalent cases 67
Table 6.4 Thrust and Torque average values and oscillation amplitude for different codes in p2.5 70
Table 6.5 Thrust and Torque average values and oscillation amplitude for different codes in p2.7 71

Chapter 1

Introduction

Energy production with renewable sources has gained a lot of attention in last years, especially after the Paris Agreement (2015), which has paved the way for a massive reduction of Greenhouse Gases (GHG) emissions, like CO2, namely the main cause of climate change. Another important aspect that has to be considered is the continuous growth of human population, to which is strictly related an increase in the energy demand. These requests can be satisfied by means of more efficient and sustainable power generation systems and renewable sources might be a consistent solution. Among these Green options one of the most effective and stable technology is wind power production, mainly in term of onshore installations but also offshore ones seems to be very promising as will be showed in one of the next paragraph, an overview about wind energy.

1.1 Historical wind energy review

Mankind is not new to wind energy, it started using it in sail propulsion for boats and the first written references about windmill concepts were already present during seventeenth century B.C. in Babylonia. The first accurate design documentation of a windmill for grain grinding dates back to 200 B.C., in Persia (a vertical axis concept was applied). Evidences of stable windmill presence in Europe date from 13th century. This kind of machines had horizontal axis design and was then exported in North America by Dutch communities in 17^{th} century. The first horizontal axis wind turbine (HAWT) designed for electricity generation was instead built in Denmark at the end of 1890 and then in the next decades the installation began spreading all around Northern Europe and United States, and the power capacity was about tens/hundreds of *kiloWatt*. Low solidity and high rotational speed rotors studies continuously improved until the cost of electricity produced from fossil fuel based plants became significantly lower (3 to 6 cents/kWh of fossil fuel plants versus 12 to 30 cents/kWh of wind ones), around 1940. This difference increased more and more, causing lack of interest and funding in wind energy studies, until 1973, when Petrol Crisis occurred. This fact plus the awareness of the possibility of exhausting the fossil fuel reserves led scientific community and policy makers to a reborn interest in wind energy. Since that time the technology constantly grew and keeps doing it today.

1.2 State of the art and forecasts

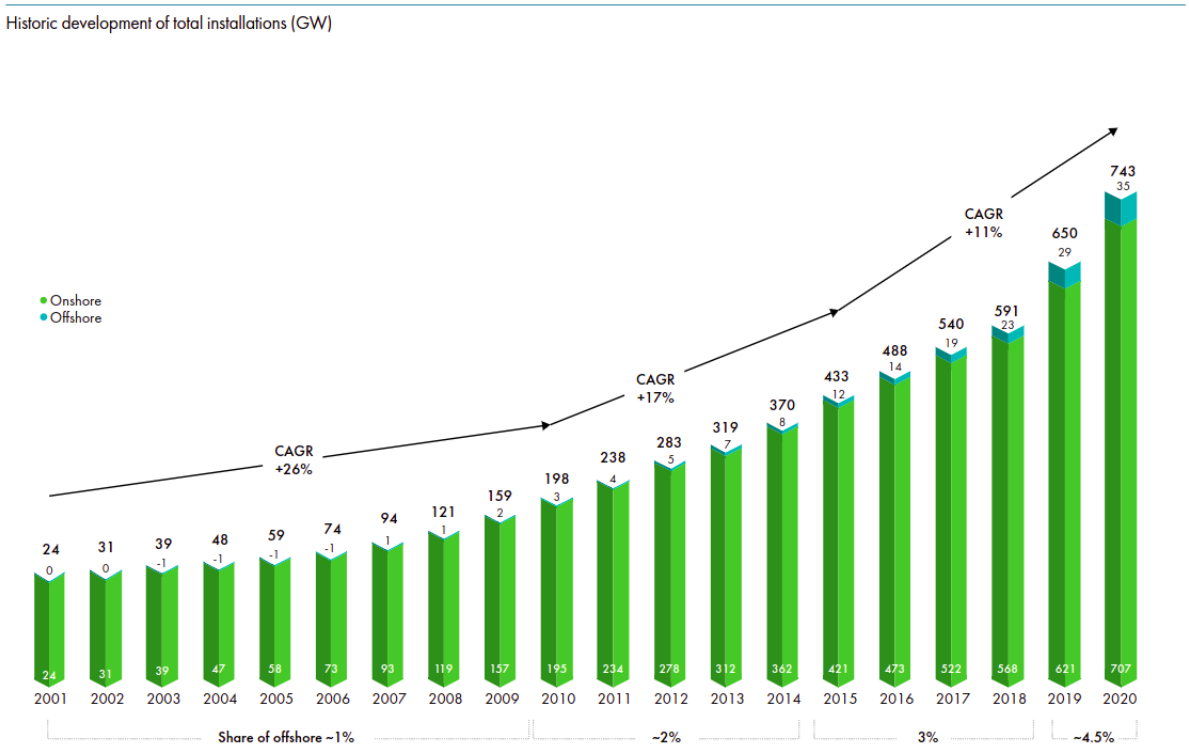


Figure 1.1. Global cumulative installed wind capacity from 2001 to 2020

Nowadays wind energy is considered, with solar one, the driving technology to achieve low carbon emissions target. The resilience of this sector is underlined in the GWEC-Global Wind Report 2021, which states that although the 2020 pandemic conditions made the global supply chain and the project constructions more challenging, wind industry showed a year-over-year (YoY) growth of the 53% , with 93GW of new installation (6.1 GW commissioned in the offshore market). Moreover in 2020 capex to wind offshore installation overtook the investment on oil and gas offshore for the first time. While onshore solution is a stable renewable solution also in terms of levelized cost of energy (LCOE), offshore one has to be tuned at art to be competitive in the market. The development of this plants and increasing energy production of the whole wind energy industry requires in parallel an adequate improvement of electricity storage capacity (in terms of batteries). Alternatively some interesting projects, like Power-to-X, are proposing Green Hydrogen production and storage on offshore platforms, which will then be transported and converted in electricity or used for propulsion purposes. The flexibility and the cost effectiveness of this solution, compared to traditional cable electricity transportation, if demonstrated, will be a key factor for a further growth of the offshore solutions in the energy market.

According to GWEC [1] the wind energy market is expected to grow on average by 4% each year in the next five ones; with 469GW installed (i.e. an average of 94GW per year). It is noteworthy that the compound annual growth rate (CAGR) for onshore will be 0.3% and of 31.5% for offshore installations, that is likely to quadruple their number.

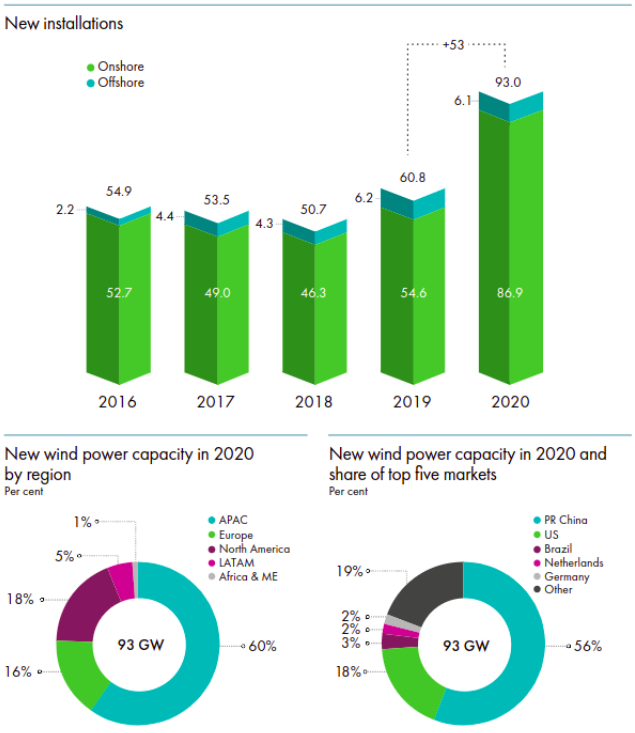


Figure 1.2. Amount of new installations from 2016-2020

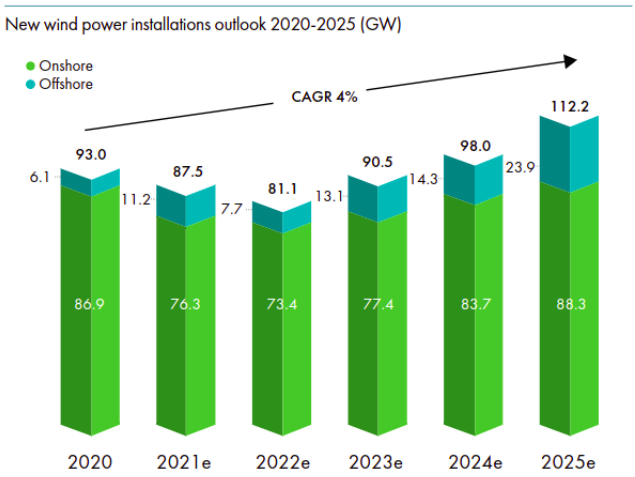


Figure 1.3. Expected wind capacity installations for the next five years

1.3 Offshore Wind Turbines

Last years rise of offshore wind turbines installations and their expected growth are related to some advantages they have with respect to their traditional counterpart: sea winds have high velocity and low turbulence intensity (TI), the size can be higher due to the fact that turbines are assembled in harbour and then transported offshore (while road transportation is a bottleneck for the maximum dimension); moreover,

both acoustic emissions and visual impact are strongly reduced once the machines are installed away from the coast.In literature this kind of machines with fixed foundations is also addressed to monopile type.
The main limitation they have is an higher LCOE (offshore cost is 90-120€/MWh while onshore one 50-60€/MWh), caused by a significant amount of investment costs associated mainly to purchase and installation of underwater substructures. Hence the appeal of these machines is limited to world regions with shallow depth seas. The actual breakthrough of this technology may be the floating offshore wind turbine (FOWT) concept, in which stiff foundations are replaced by a floating substructure anchored to the seabed by means of chains or cables. This feature cancels completely the sea depth constraint even if floaters costs are significant, even higher than rigid foundations. The three main floater concepts are presented in the figure below.

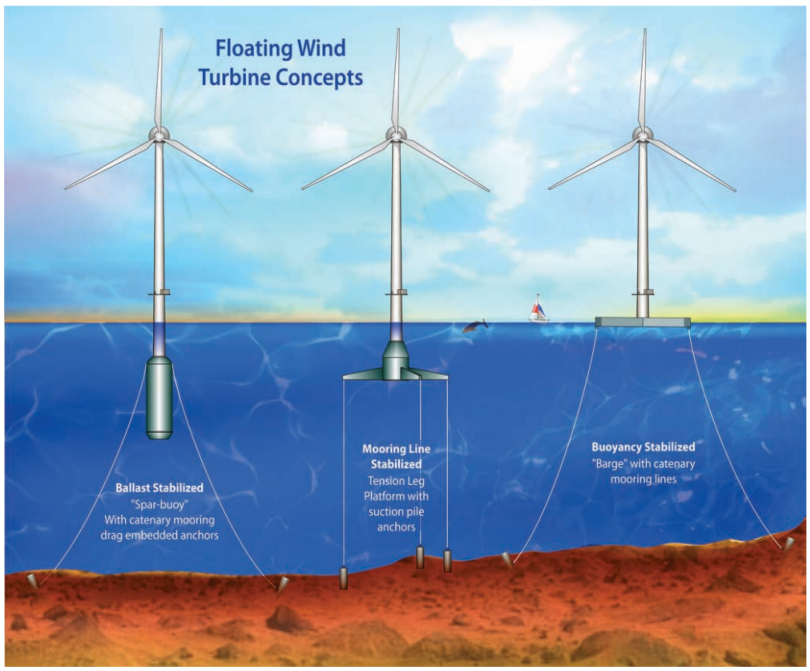


Figure 1.4. Typical floaters concepts for floating wind turbines

- **Barge**: this kind of platform uses buoyancy, achieved by its relatively large water-plane plan-form area, to maintain stability and is moored via catenary lines. From a construction and installation perspective it is the simplest and least expensive platform concept. The major penalties associated with this design are the significantly larger structural loads.

- **Spar-buoy**: this type maintains static stability by using ballast to shift the center of mass below the center of buoyancy. Catenary lines trail from the platform down to a system of anchors, maintaining station without generating large tensile loads. The ultimate and fatigue loads associated with a spar-buoy, with the exception of the tower base bending moment, are generally lower than those associated with other platform types.

- **Tension-leg platform (TLP)**: this is a mooring-line stabilized structure. Tension in the mooring-lines, running between the anchoring structures and the

over-buoyant platform structure, results in a dynamically stiff system. A number of researchers have demonstrated through simulations that a wind turbine mounted on a TLP exhibits only slight increases in ultimate and fatigue loads compared to a land-based turbine.

Until none of this platform concepts overtakes the other options, there will not be a serial production by industries of this components and costs will keep high. Furthermore one of the strategies to reduce the cost of FOWT energy is leverage on scale economy, i.e the installation of the hugest feasible machines in large offshore wind farms. Hence scientific research in this field gets paramount importance: increasing the knowledge about the hugest number of aspects will help industries working in the most profitable environments.

1.4 Research Status on FOWT

Due to the absence of fixed substructure, the aerodynamic forces are transferred from the turbine to the floater, generating a complex non linear aero-hydrodynamic coupling. As a consequence different aero-hydrodynamics codes were developed, the most popular is FAST code (NREL), which takes into account this coupling, and is based on Blade Element Method (BEM) theory; but as showed in the the next section and underlined first by studies from Sebastian and Lackner [2] when dealing with unsteady aerodynamics the BEM concept starts lacking in accuracy. Research focused then on high fidelity CFD simulations of the whole FOWT (Tran and Kim [3], [4]) under imposed motion. The drawback of a complete simulation lies on the non affordable computational costs, both in terms of computing capacity and time; hence even if this studies gave resonance to the complexity of the problem physics, research started focusing on simpler codes able to model in an accurate but less expensive way the flow-turbine interaction: Actuator Disk (AD) , Actuator Line (AL), Free Wake Vortex (FWV) methods were developed for this scope. The AL method is the one used in this thesis work and will be accurately described in next chapters. Simultaneously to numerical research experimental one developed in order to validate the codes. The aero-hydrodynamic coupling led to separated analysis of floaters and turbines in order to respectively comply Froude and Reynolds similarities. One example of this split is the presence of two different phases, I and III ([5], [6]) in the OC6 campaign program, created by NREL; Politecnico di Milano (POLIMI) is sharing the numerical results of this thesis work for the phase three of this campaign, and will provide also its wind tunnel (GVPM) experimental results for codes validation. Previous wind tunnel tests on FOWT aerodynamics were carried out here at POLIMI, namely LIFES50+ [7] and UNAFLOW ([8], [9], [10]) and the university also developed a 6 DoF pedestal to reproduce the floater motions; this system has the scope to be used in a hardware in the loop (HIL) testing strategy ([11]), in which the aerodynamic forces act like and input for the pedestal motion, to simulate the aero-hydro interaction.

1.5 Unsteady Aerodynamics on Wind Turbines

1.5.1 Two-dimensional blade aerodynamics

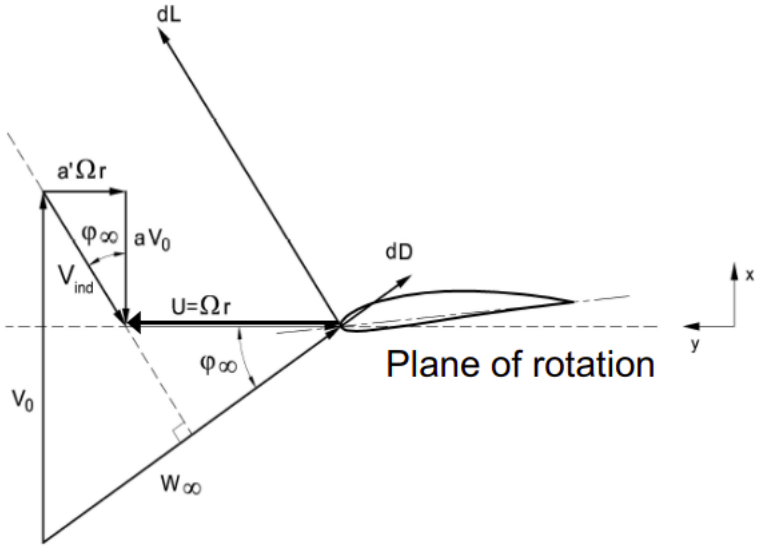


Figure 1.5. Two-dimensional velocity triangle of a wind turbine airfoil section

Before entering unsteady aerodynamics it is worth to make a review on simple 2D blade aerodynamics([12]), in fact calculation procedure of lot codes forces calculation ground on this simple schematic, one of this is the in-house POLIMI developed code. In figure 5.6 we can appreciate the average velocity triangle on a wind turbine section: a flow moving at subsonic velocity is able to feel the presence of a downstream obstacle and will redistribute in order to accomodate the presence of a rigid body shape, and this effect is quantified by the term a (namely axial induction); moreover in a wind turbine the flow is tangentially deflected by the rotation of the blade (and the deflection is accounted in a', tangential induction). The norm of the two velocities corrected with their respective inductions is the relative inlet velocity W_D, while their ratio is tangent of φ_∞ (incidence angle). The difference between the latter and the pitch or stagger angle(β_c) is the attack angle α and it is of paramount importance since the forces that the profile exchanges with fluid (and consequently the power extraction) depend directly on it. In fact every blade profile feels a force that can be decomposed in two components: lift (L), perpendicular to W_D direction, and drag (D), parallel to W_D. For each airfoil type we can define *Polars*: graphs describing the trend of lift and drag non dimensional quantities (respectively C_L and C_D) with respect to the attack angle at different Reynolds ($Re = \frac{W_d c}{\nu}$, in which c represents the airfoil chord) regimes.
BEM based codes, relying on this theory, make an iterative convergence on the Re, a and a' to extract the optimal (i.e design) chord and aerodynamic at different rotor annuli. BEM is considered a reference for the design of fixed turbines. Nevertheless this models are based on some hypothesis:

- each annular ring is independent of every other annular ring;
- steady and non turbulent flow;
- axial symmetric inflow;
- 2D sectional flow on the blades and no dynamic effects

Even if these assumptions are not fully real, at engineering level they represent an acceptable approximation of real flow-turbine interaction and give reliability to BEM codes for design and control.

1.5.2 Unsteady aerodynamics parameters

Although these BEM/Linear Momentum Theory assumptions hold with fixed turbines when dealing with FOWT it is required to consider six additional degrees of freedom for the turbine and the original flow approximations are no more fully trustworthy.

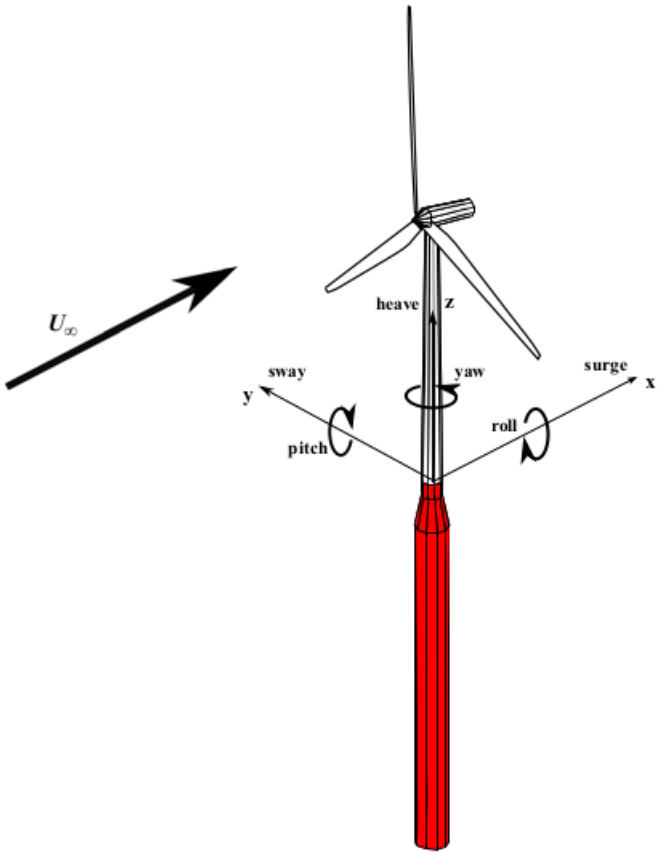


Figure 1.6. The 6 DoF of a Floating Wind Turbine

Due to the presence of the floater the stiffness associated to all the six degrees of freedom is reduced with respect to the fixed substructure case and this has an impact on the operating conditions of the turbine due to the presence of unsteady effects, caused by the hydro-wind dynamic interaction of the turbine with wind and sea waves. As roughly introduced from the floater list, each design is more or less

sensitive to a single degree of freedom and this can translate into a lower or higher level of unsteadiness which needs to be quantified with parameters, in order to accurately forecast the system response.
A first overview to understand the degree of unsteadiness of our system response can be given by a *Reduced Frequency Analysis* (k):

$$k = \frac{\omega c}{2V} \tag{1.1}$$

with

$$V = \sqrt{(\Omega r)^2 + U_\infty^2} \tag{1.2}$$

Note: Ω stands for the turbine rotational frequency while ω for the pulsation of the disturbance. k is modulated by the frequency of the excitation (which for simplicity will be considered as only one harmonic forcing, characterized by an amplitude and a frequency of oscillation). Some studies listed the value of k for which the system response becomes unsteady:

Flow Aerodynamics	Range
Steady State	0
Quasi-Steady	$k < 0.05$
Unsteady	$0.05 < k < 0.2$
Highly-Unsteady	$k > 0.2$

Table 1.1. *Reduced Frequency Analysis*

The equation of 1.1 shows how different can be the aerodynamics at the different blade spans in fact both the chord (c) distribution and the rotational speed (Ωr) are function of the radial position of the blade, showing strong problem dependency on the local position. This explains why it is so important to know accurately flow behaviour in order to operate the turbine in optimal conditions, both in terms of power output and thrust, the first is associated to energy production and the latter is strongly linked to blade loads and thus to machine fatigue life.
We have just seen how BEM steady hypothesis falls above a certain k threshold. This implies the introduction of more complex models to be implemented like *dynamic stall* (McCroskey [13], Larsen [14]), which can be added to the codes used or in the wing polars, plus, stalled flow cannot be considered two dimensional, hence another aforementioned BEM hypothesis breaks.
Another effect that several studies highlighted, like Mancini's thesis work [15], is a delay in phase of the turbine response to the motion at which it is subjected. This lag in the response, quantified mainly by thrust oscillations, caused also by the interaction of the rotor with the wake it generates, alias *Dynamic Inflow*, was accurately analysed by Mancini who tuned a first order model to take account of this effect([16]). The tuning was performed according to a quasi-static flow behaviour, since preliminary reduced frequency analysis gave as output a k in that range,and quantified unsteady

response by means of system additional aerodynamic parameters (i.e m_{aero}, c_{aero}) to estimated the delayed response.
A rough parameter used in aero-elastic theory, as remarked in Boldrin's Thesis ([17]), is the *Wake Reduced Velocity* ($V_w^* = \frac{V}{fD}$), defined as the ratio among on oscilation period (f^{-1}) and and an estimation of the time implied by a particle to drift past a characteristic length of the structure, i.e. the wind turbine diameter (D). Qulitatively, a V_w^* high value means the flow particles are not affected by the turbine oscillation (quasi-steady condition), while when the parameter reduces the interaction between the machine motion and the flow is relevant. Similarly, another adimensional number can be used to quantify the amount of disturbance introduced in the system: it is addressed as $\frac{\Delta V}{V_0}$, in which ΔV indicates the maximum absolute value reached by the rotor of a FOWT and V_0 is the free-stream velocity module. Of course each platform degree of freedom will give a different ΔV expression to the turbine rotor velocity, and will be defined in next chapters. In fact in this thesis work a comparison between pitching and surging turbine unsteady aerodynamics outputs was performed and the parameter chosen for the selection of the cases to be analysed was indeed $\frac{\Delta V}{V_0}$. The reason of the choice of the parameters will be presented in the dedicated Chapter.

1.5.3 Pitch and Surge platform motion and limits of Linear Momentum Theory

Among the six additional degrees of freedom that the turbine platform can have, the Thesis Work focused on two of them, surge displacement and pitch rotation, which are marked in literature (Huang, [18]) as the most relevant in terms of turbine performances influence. Sebastian Lackner studies ([19]), for example, evidence this fact:"Aerodynamically relevant platform DOFs that may lead to unsteady loading were identified for each of the platform configurations: pitch for the barge, pitch and yaw for the spar-buoy, and surge and pitch for the TLP. Additionally, the barge and spar-buoy floating systems were shown to have a greater fraction of unsteady flow energy because of platform motions than the TLP system."
Surge motion is a linear displacement in the streamwise direction, the rotor moves in leeward and windward direction uniformly and hence the load variation that the turbine feels can be modelled with a step forcing. The sudden load variation induces a delay in the response (Dynamic Inflow). This uniform interaction of the rotor with the flow allowed an equivalent analysis in some studies (Berger [20], Mancini [**?**], [16]): turbine is kept fixed and a steep and sudden variation of the blades pitch is applied, then the system delayed response is measured.
When dealing instead with a rotation of the turbine around a pitching axis things get an additional degree of complexity, since the rotor is skewed with respect to the incoming flow. As showed from different studies (Huang [18], Wen et al. [21], [22]) pitch has an higher impact on wake deficit and to give an idea of the complexity Huang article states that "the fluctuating range of the aerodynamic loads with platform pitching motion is almost five times that with platform surging motion under the same operating wind-wave condition". Other researches made by Persico et al. ([23]) underlined limitations of BEM based simulations comparing them with high fidelity CFD ones under high amplitude platform pitch operations, due to the detrimental

effects of the dynamic stall.

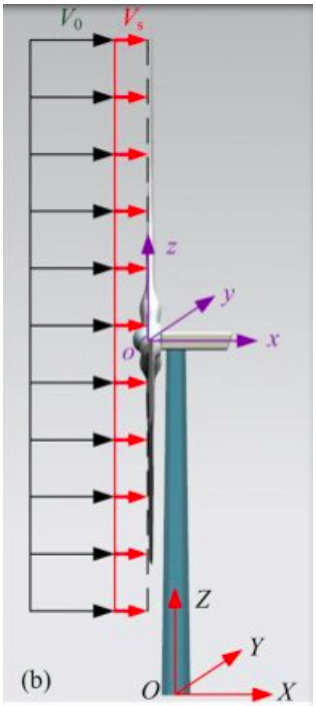


Figure 1.7. Inflow distribution for Surge motion

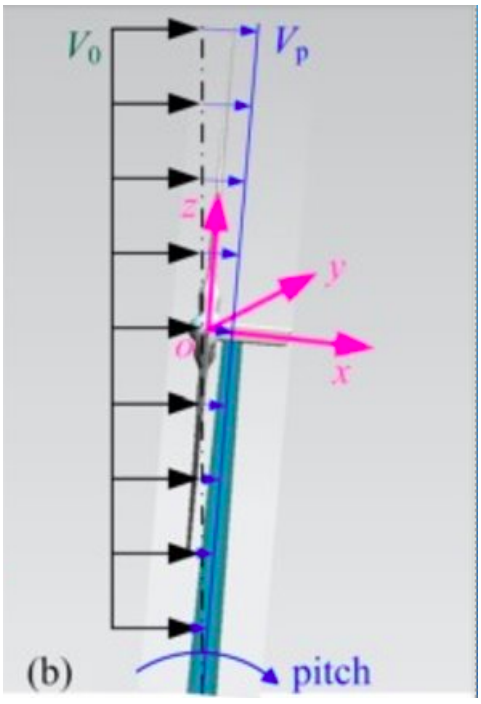


Figure 1.8. Inflow distribution for Pitch motion

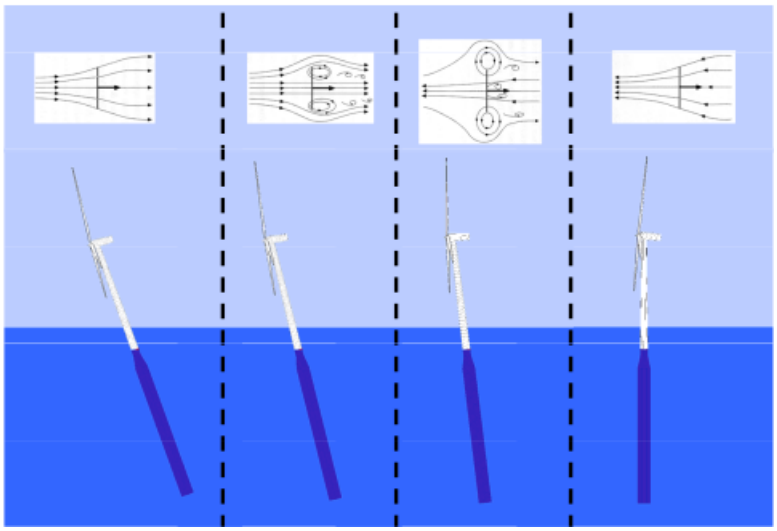


Figure 1.9. The potential flow configuration with a FOWT

Usually steady turbine operates in windmill state, but rapid drops in effective wind speed because of platform motion or drops in actual free-stream velocities result in increasing tip speed ratios (TSR). As the rotor sheds energy into the surrounding flow its behavior approaches the propeller one, as can be appreciated in the figure 1.9. This possible condition lead to a collapse of linear momentum theory on which traditional codes are based, and this happens at vortex ring state (VRS, third frame of the figure 1.9, which conditions are described for helicopters by Wolkovitch theory, [24]). Later other authors, investigated this phenomenon for wind turbines, defining and comparing the criteria chosen to predict its occurrence (Virè, [25]). In the VRS the momentum balance equations break down entirely and the thrust and torque of the rotor are driven primarily by the rate of energy dissipation in the toroidal vortex ring. Eventually, the relative wake will reverse and the wind turbine will impart energy into the flow field and behave like a propeller, most likely at the zenith of the pitching motion.The transitions heavily depend on the relative velocities between the free-stream wind and the platform kinematics.
These motions can induce limitation one the momentum balance theory on which most of the codes relies: as a wind turbine extracts energy from the flow field, the downstream flow velocity:decreases. The change in flow momentum across a wind turbine rotor may be related to the rotor thrust, T, in terms of the wake deficit (decrease in the free-stream wind far downstream of the rotor), w:

$$T = \dot{m}U_\infty - \dot{m}(U_\infty - w) = \dot{m}w \tag{1.3}$$

with:

$$\dot{m} = \rho_\infty U_\infty A \tag{1.4}$$

where ρ_∞ and U_∞ are respectively free-stream air density and velocity and A he rotor cross section. Performing an energy balance on the system, it becomes apparent that the power output (P) from the turbine can be expressed as the work done on the air by the turbine:

$$P = \dot{m}w(2U_\infty - w) = T(U_\infty - v_i) \tag{1.5}$$

in which v_i is the rotor induced velocity

$$v_i = aU_\infty \tag{1.6}$$

Momentum balance theory shows its validity if

$$U_\infty \geq 2|v_i| \equiv a \leq 0.5 \tag{1.7}$$

Once a becomes higher than 0.5 the linear momentum theory falls and this condition is called turbulent wake state. For a floating wind turbine, additional effective wind contributions on the rotor are generated by the platform motions (V_{pl}), hence a new condition must be satisfied:

$$U_\infty + V_{pl} \geq 2|v_i| \tag{1.8}$$

Note that V_{pl} , in particular for pitch and surge motions, could be both in the same or opposite versus of U_∞, leading to contrasting effects on reliability of the theory. If the inequality in equation 1.8 is not satisfied, then multiple flow directions at the rotor are possible and the momentum slip-stream assumption breaks down

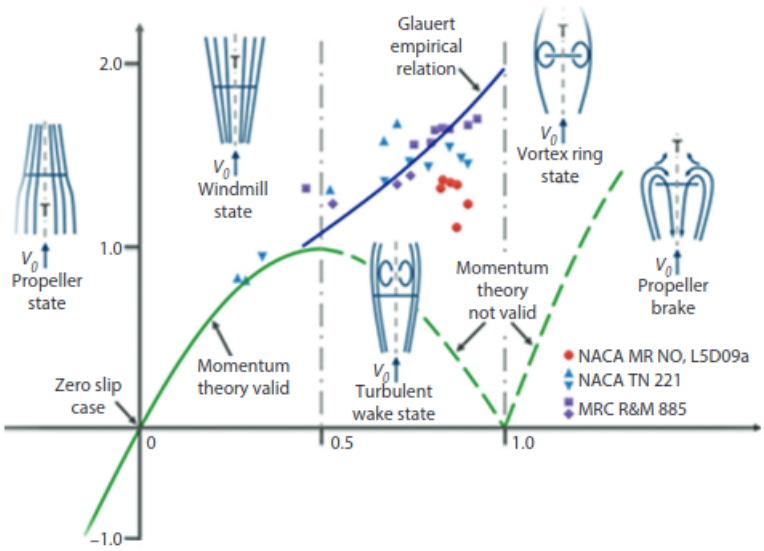


Figure 1.10. Axial induction (a) versus Thrust (T)

The figure 5.6 evidences the strong link between axial induction and thrust on the machines plus the complexity and variety of flow configurations in which a turbine can operate. When Momentum theory is no more valid the Thrust is expected to grow despite what is expected in the theory; empirical correction becomes necessary, The most famous manipulation is the Glauert empirical relation, which is implemented in BEM codes in order to obtain maximum reliability.

1.6 Thesis Scope and Themes

As previous paragraphs have spotlighted the complications that transient turbine motions can bring in terms of turbine performances estimations and possible design.

Moreover these unsteady effects seems to become more relevant with the trending desire of increasing the turbine sizes (in order to take advantage from scale economy), for example the operating conditions at the tip of two blades in a turbine under platform pitching differ more if the machine is bigger.
Consistent design and control of new FOWT strongly relies on a deep knowledge of unsteady aerodynamics, which can allow good estimation of turbine behaviour in terms of performances, loads and wake generation (the latter is important not only for single turbine analysis but also for predictions of wind farms performances). These are the reason which oriented this Thesis Work, since it focused on the comparison among Pitch and Surge effects on the turbine performances and wake generation, in order to distinguish and quantify in a clear way the single motion effects. These outputs will be useful in order to validate the consistency of the AL code for the machine behaviour prediction and description, and to enlarge the worldwide wind turbine unsteady aerodynamics database. On this basis the Thesis work evolved on different steps, listed below:

- study of the in-house developed AL code and implementation of a prescribed platform pitch motion for the turbine;
- set-up of the CFD simulation with particular attention to domain size choice;
- comparison among numerical pitch and surge platform motion simulations in terms of machine performances (i.e Thrust, Torque, Power), blade aerodynamic response (Reduced volicity & Aerodynamics parameters quantification and distribution along the span) and consequent wake generation, measuring the flow velocity profile at two different downstream position;
- comparison of AL simulation outputs with already available OC6 experimental data.
- comparison of AL pitch simulations with other numerical tools

Chapter 2

Introduction to CFD

This chapter is an introduction and overview on Computational Fluid Dynamic theory and application on the OpenFOAM software. The main references for this chapter are two, namely the Malalasekera and Versteeg book [26] and the notes from the lecture of Modeling Techniques of Fluid Machines course of Prof. Persico Montenegro [27].

2.1 CFD

Computational Fluid Dynamics (CFD) is a set of numerical methods applied to obtain approximate solutions of problems of fluid dynamics and heat transfer. It is the most recent approach in the philosophical study and development of fluid dynamics. In the 17th century the foundations for *experimental* fluid dynamics were laid in France and England. In the 18th and 19th centuries always in Europe huge progress was made in the field of *theoretical* fluid dynamics. Whoever was dealing with this wide branch of science until, say, the sixties, was dealing either with one or the other approach. The advent of modern computing made the use of numerical methods for solving partial differential equations with calculators (i.e.CFD) a powerful tool which developed between the two original approaches and is able to complement and link them.

We can briefly summarize the characteristics of the three approaches:

- The *theoretical* one gives exact solutions but can only be used for a limited class of problems;
- The *experimental* approach, if applied to full scale models, gives more realistic information. Nevertheless it is very expensive, technically difficult and affected by systematic error and uncertainities;
- CFD gives field information on the entire domain, it is relatively cheap and it allows quick changes in parameters and the simulation of conditions very difficult to be investigated experimentally. Of course it always requires verification and validation and not all sources of numerical errors can be eliminated.

With the performance-to-cost ratio of computers increasing at a continuous rate, the use of CFD is becoming less and less time and cost consuming making CFD an increasingly powerful tool. Applications can be found literally in all situations were a fluid in motion is involved, the most obvious being the aerospace industry (were it

has been applied extensively for more than sixty years now) but also, in general, in fields of engineering where the geometry is complicated or some important feature (such as the prediction of the concentration of a pollutant) cannot be tackled with standard methods.

Some of the fields where it is used are listed below:

- Automotive;
- Turbomachinery and internal combustion engines;
- Chemical processes;
- Civil engineering: fluid-structure interaction;
- Meteorology;
- Marine engineering: ship hydrodynamics, loads on offshore structures;
- Bio-medical engineering.

2.2 The steps of a CFD simulation

Any CFD simulation follows the steps that will be presented in this paragraph. Each step represent a choice that must be taken carefully and with the awareness of what it implies; it is in fact advisable to have a good background and knowledge on the physics of the problem, in order to model in the most reliable and realistic manner.

2.2.1 Mathematical Model

The mathematical model is the set of partial differential equations and of the initial and boundary conditions describing the physical system of interest. For almost all CFD problems the reference model is some convenient expression of the Navier-Stokes (NS) equations. Each model comes with a level of empiricism and simplifications that have to be taken into account when making the other choices and analyzing the results.

2.2.2 Discretization method

The discretization in a CFD simulation is both spatial (of the domain) and numerical (of the equations). The set of points or elements replacing the continuity of real space is called *grid* or *mesh*. The shape of the mesh has issues related to the numerics. This makes mesh generation the most delicate and time consuming step of a CFD simulation in most cases. Grids can be of two types, the choice depending mainly on the geometry:

- **Structured grids**: they consist of families of lines with the property that lines belonging to a same family do not cross each other and cross all the other lines only once. The advantage is that the lines of a certain family can be numbered consecutively. The structure of the solution matrix will be regular. The main drawbacks are the fact that they are not good enough for complex geometries

and that it is difficult to control the distribution of grid points (if you need a certain level of refinement in some area of the mesh all of it will be unnecessarily refined). This second problem can be solved with block structured grids allowing to conveniently subdivide the domain in sub-regions. They require special treatment at block interfaces. When they have overlapping blocks they're called *composite* or *Chimera* grids.

- **Unstructured grids**: When the geometry is complex they are a better choice. The elements can have any shape and there is no restriction on the number of neighbor elements or nodes. However the mesh generation and the pre-processing phase in general become much more complex. Also the solution matrix will no longer be regular and the solution procedure might be slower.

Discretizing the differential equations means finding a suitable method of approximating them into a system of algebraic ones describing the variables at some set of discrete locations in space and time. The most common methods (and also the most suitable for unstructured grids) are:

- **Finite Difference (FD)**: Quantities are defined over the points. The indexing is straightforward and the solution method can be optimized. The starting point is the conservation equations in differential form. At each grid point, the differential equation is approximated by replacing the partial derivatives by approximations in terms of the nodal values of the functions;

- **Finite Volume (FV)**: Quantities are defined over the volume or referred to its centroid. The indexing is variable, depending on the shape of each cell. The solution method must be adequate to any type of cell shape.This time the integral form of the conservation equations is used. Interpolation is used to express variable values at the control volume surface in terms of the nodal values. Its succes among engineers comes mainly from the fact that all the terms that need to be approximated have physical meaning.

2.2.3 Analysis of the numerical schemes

Numerical schemes establish a relation between neighboring mesh points. Any discretization will generate errors as a direct consequence of the replacement of the continuum model with its discrete representation (truncation error). Any numerical scheme must satisfy:

- consistency;
- stability;
- convergence;
- conservation;
- boundedness;
- realizability;
- accuracy.

2.2.4 Solution method

After discretization we end up with a very large system of non-linear algebraic equations that is solved most of times iteratively. The solution algorithm depends on the problem we are considering.
It can be useful to spend some words on the **PISO** algorithm since it is the one employed in the solvers considered for this work. Pressure-Implicit with Splitting of Operators (PISO) was proposed by Issa in 1986 [**?**]. In the pressure-velocity system there are two complex coupling terms: the non linear convection one and the linear P-U coupling. The idea behind PISO is that for low Co number (small time step) the linear term prevails on the linear one, making it possible to repeat a number of pressure correctors without updating the discretization of the momentum equation. PISO is very useful in simulations where the time-step is controlled by external issues and temporal accuracy is important (LES for example). In such cases, the assumption of slow variation over non-linearity holds and the cost of momentum assembly and solution can be safely avoided.

2.3 Governing equations

2.3.1 Conservation laws

CFD is based on the study of conservation laws. In considering them the concept of *flux* is very important. A physical flux F is the amount of the considered quantity entering and leaving the considered domain. We also have to consider *source terms*. They can be of two types:

- Surface sources Q_s;
- Volume sources Q_v;

The general integral form of the conservation equation of a scalar ϕ is:

$$\frac{\partial}{\partial t}\int_\Omega \phi \, d\Omega = -\oint \vec{F} \cdot \, d\vec{S} + \int_\Omega Q_v \, d\vec{\Omega} + \oint_S \vec{Q_s} \cdot \, d\vec{S} \tag{2.1}$$

We can obtain the differential form by applying Gauss' theorem to the surface integral of the flux:

$$\frac{\partial \phi}{\partial t} + \vec{\nabla} \cdot \vec{F} = Q_v + \vec{\nabla} \cdot \vec{Q_s} \tag{2.2}$$

The fluxes appear only under the divergence operator, which is the only space derivative term. If all of the space derivatives can be grouped under the divergence term the equation is said to be written in conservative form. Fluxes can be generated from two contributions:

- Convective: It represents the amount of the quantity ϕ being carried away or transported by the fluid flow.
$$\vec{F_c} = \phi \vec{U} \tag{2.3}$$
- Diffusive: It represents the amount of ϕ being carried away or transported by its gradient.
$$\vec{F_D} = -k\rho \vec{\nabla} \phi \tag{2.4}$$

If the conserved property is a vector, the flux and the surface source term become tensors.

2.3.2 The equations of fluid mechanics

Our mathematical model is completely defined by three conservation equations:

- Continuity equation;
- Momentum equation;
- Energy equation.

State quantities can be linked by an equation of state. If the given set of equation is applied to viscous flows we talk of *Navier Stokes equations*, if it is applied to inviscid flows we talk of *Euler equations*.
Continuity equation is referred to a scalar property:

$$\phi = \rho \tag{2.5}$$

Mass does not diffuse.
Momentum equation is referred to a vectorial property:

$$\phi = \rho\vec{U} \tag{2.6}$$

Momentum does not diffuse either. There are contributions to the source terms coming from external volume forces and from internal forces. The internal forces can be seen as an expression of deformability. They are represented by a tensor. If the fluid is *Newtonian* this tensor can be expressed as follows:

$$\overline{\overline{\sigma}} = -p\overline{\overline{\mathbf{I}}} + \overline{\overline{\tau}} \tag{2.7}$$

where $-p\overline{\overline{\mathbf{I}}}$ is the isotropic pressure component and $\overline{\overline{\tau}}$ is the viscous shear stress tensor:

$$\overline{\overline{\tau}}_{ij} = \mu((\frac{\partial U_j}{\partial x_i} + \frac{\partial U_i}{\partial x_j}) - \frac{2}{3}(\vec{\nabla} \cdot \vec{U})\delta_{ij}) \tag{2.8}$$

it represents the internal friction force of fluid layers against each other.
For energy equation the conserved quantity is total energy:

$$E = e + \frac{1}{2}\vec{U}^2 \tag{2.9}$$

$$\phi = \rho E \tag{2.10}$$

We can consider both types of flux:

$$\vec{F_C} = \rho E\vec{U} \tag{2.11}$$

$$\vec{F_D} = -k\vec{\nabla}T \tag{2.12}$$

The volume source terms are the work of the volume forces f_e and the heat sources:

$$Q_V = \rho\vec{f_e} \cdot \vec{U} + q_H \tag{2.13}$$

The surface sources are the result of the work done by the fluid by the internal shear stress acting on the surface of the control volume:

$$\vec{Q_S} = \overline{\overline{\sigma}} \cdot \vec{U} = -p\vec{U} + \overline{\overline{\tau}} \cdot \vec{U} \tag{2.14}$$

Finally, the full set of Navies Stokes equation can be written as follows:

$$\frac{\partial \rho}{\partial t} + \vec{\nabla} \cdot (\rho \vec{U}) = 0 \tag{2.15}$$

$$\frac{\partial (\rho \vec{U})}{\partial t} + \vec{\nabla} \cdot (\rho \vec{U}\vec{U} + p\overline{\overline{\mathbf{I}}} - \overline{\overline{\tau}}) = \rho \vec{f_e} \tag{2.16}$$

$$\frac{\partial \rho E}{\partial t} + \vec{\nabla} \cdot (\rho H \vec{U} - k\vec{\nabla} T - \overline{\overline{\tau}} \cdot \vec{U}) = \rho \vec{f_e} \cdot \vec{U} + q_H \tag{2.17}$$

2.4 Turbulence modeling

Our mathematical model appears to be rather complex on multiple levels. It is a set of fully coupled time dependent non linear partial derivative equations. Non linearity comes from compressibility and from the convection term which is responsible of *turbulence*. This makes these equations not integrable analytically if not in a few cases with very limiting hypothesis. Our issues are not only mathematical but also physical. In a great number of engineering applications we have to deal with turbulance that is a very complex and chaotic phenomenon. Early theories on its origin have been formulated by Landau (1944) and Lorenz (1963). Landau's theory came from a linearized analysis of the equations and has proved wrong. Lorenz was a metheorologist and he came out with the concept of *deterministic chaos*, a feature emerging from non linearity. He suggested that turbulence might be considered as a form of deterministic chaos. Instead of looking for a definition, for a better understanding of what we mean when we say that a flow is turbulent, it can be useful to list some characteristics:

- Fluctuations (both in time and space) even in presence of time invariant boundary conditions;
- Non linearity: turbulence is caused by the non-linear nature of Navier-Stokes equations and it is triggered for sufficiently high values of non-linearity parameters (like Reynolds); turbulence is in fact the final state of a process composed by a succession of non-linear instabilities, called transition;
- Vorticity: related to the concept of *eddies*, highly unstable streaks and swirls, the flow undergoes distortion and spin, with sizes ranging from the problem scale to very small scales and are then dissipated by molecular viscosity (broadband oscillations);
- Dissipation: momentum is transferred to smaller eddies in an inviscid way down to a certain scale, in which viscous dissipation occurs;
- Diffusivity: the chaotic motion enhances mixing via transport, resulting in mass, momentum and energy diffusion rates much larger then just the molecular ones;

Because of its nature the description of turbulence strongly relies on statistical methods and tools as ensemble average, cross correlation and auto correlation allowing us to recognize time and length scales of turbulence. One of the core of modeling techniques grounds indeed on the identification of turbulence length scales and dimensional analysis in order to simplify the number of variables and unknowns.

2.4.1 Energy cascade and implications on CFD modeling

The energy cascade theory was proposed by Kolmogorov in 1941. The large eddies are unstable so they break up and in doing that they transfer energy to smaller ones and the process goes on. In relatively large eddies, this process is merely an exchange of kinetic energy and an indicator of this phenomenon is their Reynolds number, which is estimeted far above unity. It is useful to remind what Reynolds group represents, or else the ratio among flow inertial forces and viscous ones, and in fact at high levels the flow is dominated by its own inertia, while ,when it approaches toward value of one, the eddy scale reaches the smallest one (also called Kolmogorov scale) the kinetic energy is converted into internal energy by means of viscous dissipation, because the two forces are comparable
Based on how we treat turbulence and more specifically on which scales we choose to simulate and which ones to just model we can distinguish between:

- **Direct Numerical Simulation (DNS)**: simulating all scales of turbulence;
- **Large Eddy Simulation (LES)**: simulating the large scales and modeling the small ones;
- **Reynolds Averaged Navier Stokes (RANS)**:solving time averaged equations obtained using statistic operators.

In between LES and RANS,combining the most favourable aspects of the two stand Detached Eddy Simulation (DES) Going from DNS towards RANS accuracy but also computational cost significantly reduces. Other than on these aspects, however, the choice depends on the particular application.

2.4.2 LES: Large Eddy Simulations

It is appropriate to spend a few more words on LES since most of the CFD codes simulating HAWTs (included the AL codes used in this Thesis Work) rely on them. The main reason is that the actuator approaches that will be presented in the next chapter do not require a very fine grid around rotor blades, allowing a significant reduction of the computational cost normally required for a LES simulation, moreover, full 3D simulations employing RANS have failed to capture stall behavior [28] .
The main concept behind the use of LES is filtering. Every filter has a length scale associated with it, Δ. Rougly speaking we can say that the large eddies are the ones bigger than Δ. When the incompressible NS equations are filtered the result is something very similar to RANS:

$$\frac{\partial(\rho\overline{u_i})}{\partial t} + \frac{\partial(\rho\overline{u_i u_j})}{\partial x_j} = -\frac{\partial \overline{p}}{\partial x_i} + \frac{\partial}{\partial x_j}(\mu(\frac{\partial \overline{u_i}}{\partial x_j} + \frac{\partial \overline{u_j}}{\partial x_i})) \tag{2.18}$$

Since the continuity equation is linear, the filter leaves it unaltered:

$$\frac{\partial(\rho\overline{u_i})}{\partial x_i} = 0 \tag{2.19}$$

It is important to observe that

$$\overline{u_i u_j} \neq \overline{u_i}\ \overline{u_j} \tag{2.20}$$

so the quantity on the left side of equation 2.20 is not easily computed and the difference between the two sides of the inequality needs to be modeled:

$$\tau_{ij}^s = -\rho(\overline{u_i u_j} - \overline{u_i}\ \overline{u_j}) \tag{2.21}$$

τ_{ij}^s is called the *sub grid-scale Reynolds stress*. The name stress more than to its physical nature refers to the way in which it is treated. It really represents the large scale momentum flux caused by the action of the small or unresolved scales. The models used to approximate it are called *Subgrid-scale (SGS)* or *sub filter-scale* models. The most commonly used SGS model is the one proposed by **Smagorinsky** (1963). It is an eddy viscosity model. These models start from the assumption that the main effects of the SGS Reynolds stress are increased transport and dissipation. These phenomena are related to the viscosity in laminar flows. It is interesting to note that Smagorinsky turbulent viscosity calculation is calulcated by means of the mean deviatoric tensor instead of the rotation rate one, and this model did not gain immediately success in its years (1963), since at that time with was only possible to simulate RANS simulations, which have to deal with a huge range of eddies scales. This model was discovered later, mainly with the application of LES, in which not all scales of turbulence are modelled but just the smallest ones, and this translates in an higher rate of dissipation which is present at these shorter sizes.

$$\tau_{ij}^s - \frac{1}{3}\tau_{kk}^s\delta_i j = \mu_t(\frac{\partial\overline{u_i}}{\partial x_j} + \frac{\partial\overline{u_j}}{\partial x_i}) = 2\mu_t\overline{S}_{ij} \tag{2.22}$$

where μ_t is the eddy viscosity and $\overline{S}_{ij}$ is the strain rate of the large scale. This model can be derived in many ways. The form of the SGS eddy viscosity can be expressed as:

$$\mu_t = C_S^2\rho\Delta^2|\overline{S}| \tag{2.23}$$

where C_S is a model parameter to be determined, Δ is the filter length scale and $|\overline{S}| = (\overline{S}_{ij}\ \overline{S_i j})^{\frac{1}{2}}$. Most of these methods apply only to isotropic turbulence for which $C_s \approx 0.2$.

2.5 OpenFOAM

OpenFOAM is a free, open source computational fluid dynamics (CFD) software package based on C++. It has a large user base across most areas of engineering and science, from both commercial and academic organisations. It has an extensive range of features to solve anything from complex fluid flows involving chemical reactions, turbulence and heat transfer, to solid dynamics and electromagnetics.

It is distributed with a large set of pre-compiled applications but it is possible for the user to add his own.

2.5.1 OpenFOAM implementation of NS equations

C++ is an object oriented programming (OOP) language. This makes writing and reading the equations very simple. NS equations:

$$\frac{\partial(\rho\vec{U})}{\partial t} + \nabla \cdot \rho\vec{U}\vec{U} - \nabla \cdot \mu\nabla\vec{U} = -\nabla p \tag{2.24}$$

can be written as:

```
solve
(
    fvm::ddt(rho, U)
    +fvm::div(phi, U)
    -fvm::laplacian(mu, U)
    ==
    -fvc::grad(p)
);
```

where is it incredibly easy to recognize the terms in equation 2.24. When considering LES, equation 2.24 becomes:

$$\frac{\partial\overline{\vec{U}}}{\partial t} + \nabla \cdot \overline{\vec{U}}\ \overline{\vec{U}} + 2\nu_{eff}\nabla(\nabla\overline{\vec{U}} + (\nabla\overline{\vec{U}})^T) = -\nabla\overline{p} \tag{2.25}$$

where ν_{eff} is the effective viscosity and can be expressed as:

$$\nu_{eff} = \nu + \nu_{SGS} \tag{2.26}$$

Equation 2.25 in OpenFOAM is translated as:

```
solve
(
    fvm::ddt(U)
    +fvm::div(phi, U)
    +turbulence->divDevReff(U)
    ==
    -fvc::grad(p);
);
```

where the term `turbulence-> divDevReff(U)` contains the SGS model. The open-source work environment of OpenFOAM gives a lot advantages in terms of possibility and easiness of implementation pf additional code parts and routines, this framework led indeed to the implementation and modification of the PISO solver in the AL model applied in this work, which added a volume force inside the cells to take into account the turbine blades equivalent forces, as will be presented accurately in the dedicated chapter.

Chapter 3

OC6 Program Phase III Definition

As already introduced, the AL code implementation developed in this Thesis Work have the objective of allowing the a comparison of the unsteady aerodynamics simulation results with the experimental ones defined for the OC6 experimental campaign. In fact the scope the NREL OC6 Phase III program is to validate the accuracy of aerodynamic load predictions by offshore wind modeling tools for a Floating Offshore Wind Turbines (FOWT) as it experiences large surge translation motion. Nevertheless there emerged the possibility of analyzing also numerical simulations predictive capacity also in case of platform pitch motion of the turbine. In fact it was decided to carry out experiments in the POLIMI wind tunnel (Galleria del Vento Politecnico di Milano (GVPM)) to measure the effects of the turbine rotation around a pitch axis and to add these results to the project database. This is the reason why this kind of turbine prescribed motion needed to be implemented in the AL in-house developed code. The aim of this program is thus to compare a large number of outputs from several tools and experiments in order measure and improve the knowledge on the behavior of these floating machines. For this purpose, a definition document was drafted, in order to target a common ground for the program participants and have thus consistent comparisons ([6]).

3.1 Experimental campaign set-up

3.1.1 Overview of the experimental set-up

The dataset that will be used in Phase III was developed in the UNAFLOW (Unsteady Aerodynamics for FLOating Wind) project,which was a collaborative effort under EU-IRPWIND and included Politecnico di Milano (POLIMI), ECN (now part of TNO), University of Stuttgart, and the Danish Technical University (DTU) (Bernini et al. [9]). The testing was essentially a repetition of similar testing performed as part of the LIFES50+ project [7], but with a stiffer tower and a 5° tower tilt to make the rotor vertical (offsetting the rotor tilt). The flexible tower in LIFES50+ created issues in the measurements, making validation difficult. This testing was conducted in the Politecnico di Milano wind tunnel (GVPM) – Boundary Layer Test Section, which has dimensions: 13.84 m wide x 3.84 m high x 35 m long. The air density in the facility can be considered equal to $1.177 kg/m^3$ during the test campaign. The tests

were performed using an empty inlet configuration (i.e. without roughness elements or turbulence generators) for a constant inflow velocity and a resulting turbulence intensity (TI) around 2%. Two rows of seven 2x2 m independent fans were used, producing a total fan power of 1.4 MW.

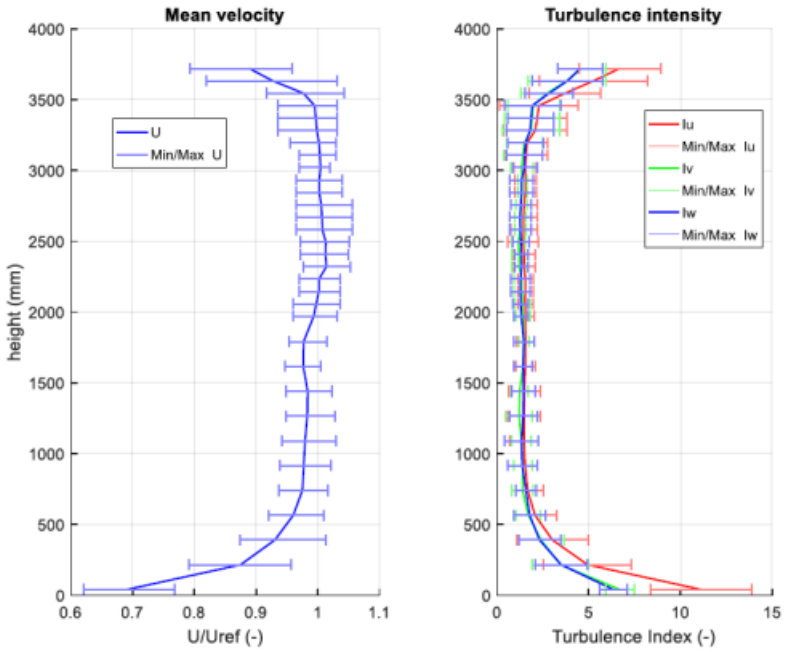


Figure 3.1. Wind profile transverse measurements output

3.1.2 Scaled Turbine design

The test turbine is a 1:75 scaled version of the DTU 10-MW RWT (Bak et al. [29]), which was designed by POLIMI within the LIFES50+ project. It is a variable speed wind turbine with individual blade-pitch control. The scaling approach for the turbine sought to preserve the thrust (C_T) and power coefficients (C_P) of the full-scale design, while scaling the overall physical dimensions by a 1:75 factor. The structural design itself was driven by mass and stiffness scaling. To achieve the scaling objectives, the wind velocity was scaled by a factor of 3, and a low Reynolds number (Re) airfoil (SD7032) was used since the Re number in the wind tunnel was 225 times smaller than full scale. The chord and twist distributions were altered to achieve the correct thrust forces and also match the torque adequately up to rated conditions (Bayati et al. [30]). The turbine was mounted on a test rig with two hydraulic actuators at the tower base to enable forced motion of the turbines scaled model in the surge and pitch directions. The overall properties of the scaled turbine are given in the following Table, 3.1.

Turbine Model Parameter	Value
Rotor Diameter [m]	2.38132
Blade Length [m]	1.10166
Hub Diameter [m]	0.178
Hub Position [m]	2.086
Tilt Angle [°]	5
Coning Angle [°]	0
Tower Diameter [m]	0.075
Distance from Pitch Axis [m]	1.6057

Table 3.1. *Scaled Turbine Model properties*

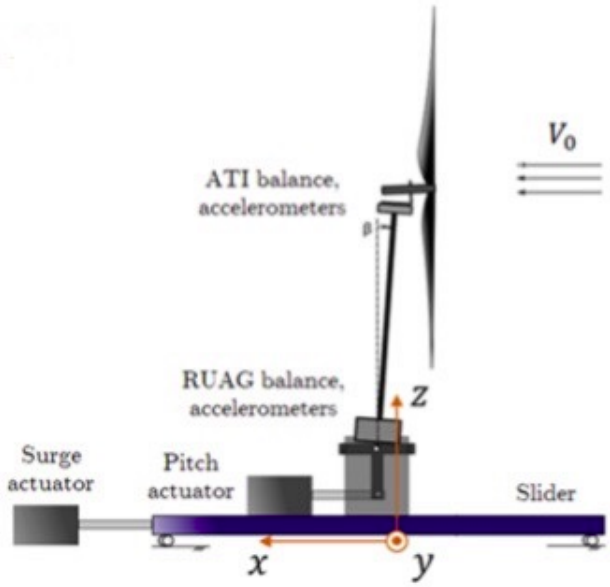


Figure 3.2. DTU 10 MW Wind turbine scaled model schematic

3.1.3 Wake measurements system set-up

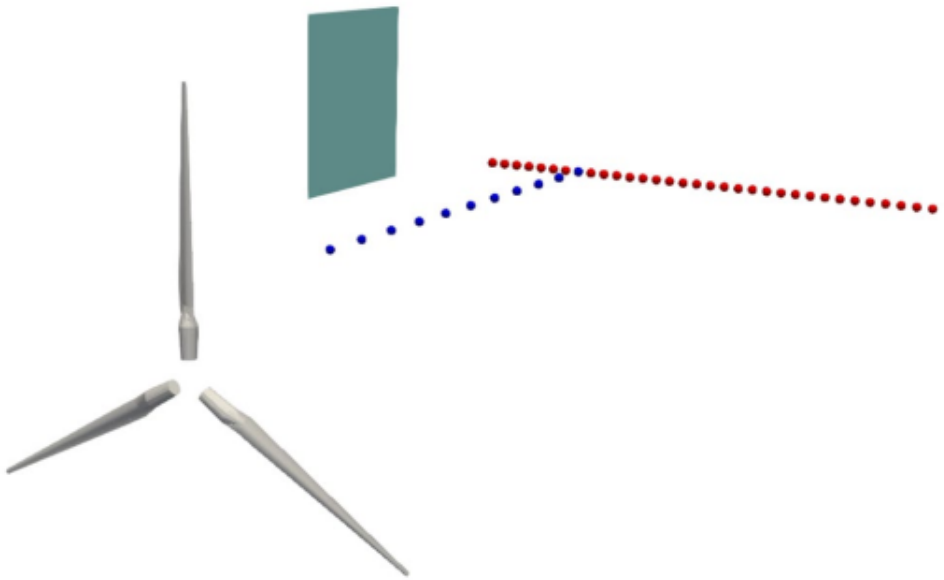


Figure 3.3. Schematic of the wake measurements set-up

As can be appreciated from figure 3.4, particle image velocimetry (PIV, green rectangle) and hot-wire measurments (both red and blue dotted lines) are required for the analysis of the wake generated by the turbine both in steady and unsteady cases, as indicated POLIMI articles [31], [32]. For steady load cases, a file will be output at each of the azimuth angles $\Phi = 0$ to $\Phi = 120$ with a $15*$ step, and from $\Phi = 120 \circ$ to $\Phi = 360$ with a 30 step (17 total azimuth angles). For unsteady cases, the PIV measurements at two turbine azimuthal positions shall be outputted, at distinct locations in the surge oscillation cycle. A hot-wire probe was set up to measure all three wind components (u, v, and w). Measurements were taken while the probe traversed the cross-wind direction (y) and along-wind direction (x). For the cross-wind motion (red dotted line), the probe started at the hub location in height, but 2.3D downstream (5.48 m), and moved from -1.6m to 1.6m in the y-direction with a spatial discretization of 100mm (33 points). For the along-wind motion(blue dotted line), the probe started with a 0.9 m offset in the y-direction, and moved between 0.9 and 2.3D (2.18 – 5.48 m) in the x-direction (PIV is from 0.61-1.36 m), with a discretization of 330 mm (11 points).

3.2 Turbine blades update

An improved blade database, with respect to UNAFLOW one, is available in order to take into-account the rotational augmentation correction for 3D delayed stall effect (thanks also to Fontanella et al. work [33]) in the airfoil polars for the numerical tools which do not already take care of it within the code (like the AL in-house developed one). The airfoils used for the scaled turbine are a 10% thick version of the Selig Database SD7032 airfoil, which is a low Reynolds number airfoil that achieves good characteristics in the Re=100k-200k range. Developing a scaled rotor

that can accurately represent all phenomena correctly is difficult, and thus the blades in the rotor were built to be rigid to eliminate the difficulty in scaling the aero-elastic behavior. Separate 2D testing was performed to characterize the airfoils used in the scaled experimental campaign. The airfoil polars were developed from these 2D tests (which can feed lifting line codes). The lift coefficients are obtained from pressure taps at the model midspan, and the drag coefficients from a wake rake.
Polars for the scale model blades have been provided at 20 radial stations, as summarized in a MATLAB file which contains a 3-D array with size 128x15x20. The 15 columns correspond to the angle of attack in degrees, and 7 sets of lift and drag coefficients for Reynolds numbers equal to: 5.0E4, 6.0E4, 7.5E4, 1.0E5, 1.5E5, 1.7E5, and 2.0E5. The third dimension corresponds to the 20 radial stations from blade root to blade tip for which these kind of polars were calculated. It has to be noted that not all codes include interpolation routines according to the Re regime at each span (like POLIMI's AL), thus OC6 provided a table to highlight the operating condition Reynolds range for the 20 radial stations.

	Reynolds						
Station	5.00E+04	6.00E+04	7.50E+04	1.00E+05	1.50E+05	1.70E+05	2.00E+05
1							
2							
3							
4							
5							
6							
7							
8							
9							
10							
11							
12							
13							
14							
15							
16							
17							
18							
19							
20							

Figure 3.4. Table for Reynolds regime prediction for each radial station

To have a comparison among the different tools, different output files are required with blade aerodynamic response at different span positions. The required outputs are: the normal force per unit length (normal to the rotor plane, positive along the wind direction), the tangential force per unit length (tangential to the rotor plane, positive in the direction of rotation), the relative wind speed (including wind inflow, rotor rotation, and induction), the angle of attack, the lift coefficient and drag coefficients.

3.3 Load Cases definition

For the program several study cases are present, starting from two steady wind conditions, namely RATED2 and ABOVE. In previous campaigns (UNAFLOW, LIFES50+) also a RATED1 condition was analyzed, but it seemed to have bad

signal-to-noise ratio (since v0=2.5 m/s) and small inflow amplification. Hence for simplicity, from now on RATED2 condition will be indicated just as RATED.

Laad Case	V_0 [m/s]	Rotational Speed [rpm]	TSR [-]
RATED(LC1.1)	4	240	7.48
ABOVE(LC1.2)	6	265	5.51

Table 3.2. *Steady Load Cases*

For the unsteady surge wind cases (namely LC2.X), the wind system was forced to oscillate in the surge direction using varying frequencies and amplitudes under the same wind conditions as the fixed ones. Each load case should be simulated for a sufficient amount of time to assess the average, steady-state conditions; hence it is necessary to consider the time to eliminate any initial transient responses, but also to average out variations due to rotor position in relation to surge motion. For the program a list of eleven surge cases is proposed (see the Appendix). For what concerns instead pitch load cases an orientative matrix of simulation was drafted by POLIMI group (Appendix). More detailed decision will be taken on which cases run and compare with the experimental results. In fact the experiments on this motion will be performed in the current year, after this Thesis Work conclusion, and the aim of the work is to compare the effect of pitch and surge motion on turbine performances; that's why a criterion was adopted to choose among all the load cases options proposed in the program even if the AL code and simulations pre-processing were set in order to generate all the outputs required by the OC6 Phase III comparison.

Chapter 4

Actuator Line Model

4.1 Overview on AL

The numerical simulations performed during this work exploited an Actuator Line based code, which was implemented at POLIMI by Schito, [34]. The reason why codes like this one are continuously developing in the numerical simulations field is the necessity of describing accurately the wind turbine and/or wind farm performances and their wake generation, with a complete but computationally sustainable modelling. In fact a complete CFD simulation of a full scale wind turbine has to deal with very different lengthscales: the domain size has to avoid the presence of spurious pressure reflections at the inlet boundary and guarantee the far wake condition at the outlet (i.e pressure at the outlet has to be equal to the ambient one). The reference length scale of this problem is the turbine rotor one (D), and the inlet and outlet boundaries are located several rotor distances from the machine (respectively, as a rule of thumb, 5/10D upstream and 10/100D downstream). Nevertheless a precise blade aerodynamics description requires an accurate discretization of the blade boundary layer (BL), which in turn means structured O-grid around each solid part with a first layer size that guarantees y+ equal to one in case of a k-omega-SST RANS simulation, the benchmark of RANS turbulence models. This restriction implies the presence of cells in the order size of the millimeter, moreover the BL cells clustering shape has to adapt to the different airfoil type along the blade span. In addition to this complex meshing task, simulation time step, to be compliant with the first layers cell, becomes very low and made computational costs unsustainable, both in research field and industry one.
Actuator Line codes gained ground on the main drawbacks of a fully resolved CFD simulations: as will be described in the next paragraphs in this kind of simulations the actual wind turbine is not present, only its corresponding forces are applied to the flow and this means that there is no more need of meshing complex shapes and BL but the simulations are run in a simple hexahedral cells grid (almost structured,or at least block structured). Moreover the AL codes have the advantage, as will be further explained in the next paragraph, of reproducing the low solidity of wind turbines rotors and thus they can describe in an accurate way the vortex system creation at blade tip and also the wake generation process, with respect to traditional BEM and AD models, which work fine only if a tip correction is introduced (like Glauert's one).

Moreover the consistent output of AL simulation of Mancini's thesis([**?**]) with the UNAFLOW experimental campaign plus a good capability in describing the wake evolution downstream the machine pushed toward the analysis of other platform motion AL simulations to be compared with the surge ones with this tool in this thesis work. A note: even if at a first look AL model does not require a mandatory tip correction, there emerged the possibility of implementing it since To achieve accurate blade loadings at the tip regions of the blades, the width of the projection function needs to be narrower than the local chord lengths, as evidencenced in recent articles ([35], [36]).

4.2 Actuator Line Description

Sorensen and Shen, [37], [28] introduced the AL model in order to overcome the aforementioned fully resolved CFD mesh scales problem. The code is integrated in a CFD environment but has the advantage of introducing only three lines, namely actuator lines, instead of a complete machine int he meshed domain. The presence of the turbine is felt in the flow by the application of equivalent blade aerodynamic forces on the three actuator lines. The resulting hexahedral mesh and the absence of blade BL pushed toward high fidelity LES, in order to better capture the flow features, especially in the wake and hence to have the most accurate reproduction of the real flow behaviour; in fact LES approach has some drawbacks in BL computations, especially if the has to deal with airfoil BL. The AL model is based on the addition of a source term in the original non compressible Navier-Stokes momentum equation, $\mathbf{f(x,t)}$, the so colled *volumeForce*.

$$\rho\frac{\partial(\vec{U})}{\partial t} + \nabla\cdot\rho\vec{U}\vec{U} - \nabla\cdot\mu\nabla\vec{U} + \mathbf{f(x,t)} = -\nabla p \tag{4.1}$$

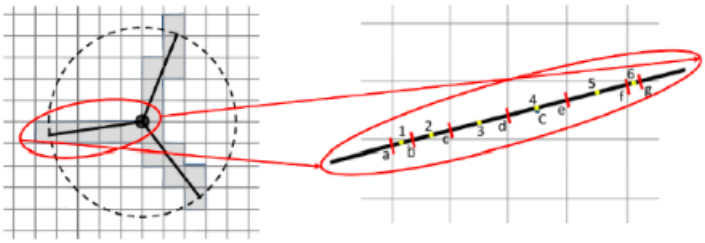


Figure 4.1. Acutator Line-mesh intersection schematic

As can be observed from figure the model each actuator line is divided in segments according to their intersections with the CFD mesh and each portion is treated as a 2D section airfoil, like a traditional BEM code does. On each of them the volume force is computed at the midpoint of the mesh-line intersection and at each simulation time step. It is worth noting that the code has an explicit formulation, i.e. the calculation of the force at a certain time step it is based on flow inlet velocity vector evaluated at the previous instant, and this is suitable for the solver used during the computation, called HAWTsolverPiso230, which is basically a PISO solver, which fits perfectly

with explicit methods, taking into account the presence of the volume force **f**. An important feature implemented in the codes is the presence of a smearing function, namely regularization kernel, η_{RK}, which has been introduced in order to avoid the presence of non physical oscillations of forces on the actuator line. The regularization kernel consists, at least in POLIMI code, in a bi-variate normal distribution, which multiplies with a convolution product the volume force applied on the single actuator point and distributes it among neighbouring cells, as can be appreciated in figure below.

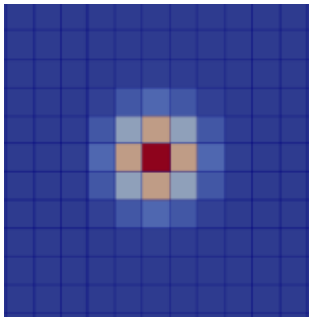


Figure 4.2. Regularization Kernel effect on neighbouring cells

$$\mathbf{f(x,t)} = \eta_{RK} \otimes \frac{\mathbf{L+D}}{V_{cell}} \tag{4.2}$$

with regularization kernel equal to:

$$\eta_{RK} = \frac{1}{2\pi\epsilon^2} \exp -\frac{-d^2}{2\epsilon^2} \tag{4.3}$$

and Lift and Drag absolute values:

$$|\mathbf{L}| = \frac{1}{2}\rho c h C_L(\alpha) W_D^2 \tag{4.4}$$

$$|\mathbf{D}| = \frac{1}{2}\rho c h C_D(\alpha) W_D^2 \tag{4.5}$$

For what concerns η_{RK} parameters: d is the distance between a neighbouring cell center and the one of the actuator point, and ϵ is the kernel size, which imposes the degree of decay of the force smearing. The choice of ϵ value is of crucial importance and impacts on the reliability and stability of the code, thus a lot of care has to be given in order to avoid wiggles in the solution and in the next chapter a deepening on this value selection will be presented. Lift (L) and Drag (D) force, as can be appreciated from formulas, are functions of airfoil profile chord and blade aerodynamic coefficient C_L and C_D (in turn functions of the angle of attack α). These parameters are given to the code by means of tabulated values of wind turbine blade profile polars, of course these quantities are assigned for a limited number of blade span positions and thus, whether the actuator point span position is not tabulated, the values are linearly interpolated between upper and lower allocated airfoil properties. It is noteworthy that wind turbine blades do not work in self similarity with respect to the Reynolds number, this means that C_L and C_D show different trends with respect

to α at different turbulent regimes, but the developed code is not able to take into account this factor, hence among all the polars made available by OC6 program, were extracted only the ones at the Reynolds which the blade profile is estimated to work. Another important parameter for the forces is the velocity triangle W_D, both for the module of the forces and the angle of attack, which defines $C : L$ and C_D values. About the sampling strategy different ways of thinking arose, one of the most popular is local sampling one (SOWFA), in which the velocity is measured at the very same point of force application: this choice relies on the idea that the line is coincident with blade pressure center, at which the bound circulation is null. Even if reliable this approach shows some limitations: the discrete nature of the domain grid may place the actuator point in position slightly different from the actual pressure center (which in addition can be be only estimated and can vary according to flow field condition); since the region of force application is characterized by strong flow gradients and turbulence, a little bit of disalignement of the sample positioning can bring to inaccurate estimation of the attack angle and thus of the forces. An alternative technique was proposed by Schito, and tuned in the thesis work of Bernini and Caccialanza [38], called Effective Velocity Model (EVM), which is implemented in the thesis code. The idea behind this approach is that in low solidity rotors, like in an HAWT, the blades can be considered as isolated profiles which depend only on upstream conditions, which in turn are altered by the induction field generated from the blades. To overcome the turbulence presence Schito proposed to place the sampling line slightly ahead of the blade and to average for each actuator point different velocities on the sampling line. The tuning was performed by Bernini and Caccialanza on the two EVM setting parameters: *upstream distance* and *sampling line length*. This method can be seen as a sort of halfway solution between local sampling approach and BEM free-stream velocity: in fact as in the latter velocity is measured upstream the rotor but as the first it takes into account non homogeneity of the entering velocity, allowing it to be suitable, up to now, for unsteady simulation. In fact, always here in POLIMI, an AL codes featuring EVM has been developed for VAWT (which are considered "unsteady by design" machines), and has shown robustness of the method (Melani [39]). Last relevant aspect of the EVM is the fact that it follows an explicit logic method which is coincident with AL method one, thus it does not add too much computational effort to the solver routines.

4.3 Actuator Line Code Structure

In this section there is a deeper description of the in-house developed AL codes, in this way it will become easier to understand its framework and how to manipulate it in order to add implementation like the one performed for this Thesis work, i.e. the rotation of the turbine rotor around a pitching axis to simulate platform pitch motion of a FOWT.The code is written in C++ in this way it is compatible and exacutable in environment like OpenFOAM. It is composed by four classes, namely HAWTurbine, HAWTbladeLine, actuatrForceLineBaseClass and actuatorForceClass. They all feature similar structure, for example constructor-destructor, getter function and writing dictionaries, a header file in which the class is declared according to the quantities and parameters useful for the completion of its required tasks, libraries for

the sampling of the effective velocity, for evaluation, application and regularization of the forces. As one can guess from the classes names they represent the AL model interaction with the flow at different level of scale:

1. **HAWTurbine** is the higher degree class since it is representative of the whole turbine (no nacelle neither turbine tower are implemented in the code), the motion law of the turbine is implemented at this level as explained in next section. At this layer force are calculated at integral level: its output are the thrust, torque and power that the rotor extract from the flow, i.e. the full scale wind turbine performances. This class getter function interacts primarly with the constant utilities of the simulation case (which give input like the rotor blades number and their hub and tip radius, rotation velocity, the rotor yaw, blade pitch and the identificatio of the airfoil polars of the blades);

2. **HAWTbladeLine** is one degree below the turbine and it is related to the single blades forces calculation according to the the EVM velocity sampling. In this class there are dictionaries aimed to the interpolation of the Cl and Cd coefficients along the span length and to the coordinate transformation from absolute domain cartesian frame of reference (x,y,z) to the blade relative one (tangential, axial,radial) and viceversa. There is a dictionary for post processing purpose in which it is possible to decide which outputs have to be written, for each blade, like the number of actuator points, their positions, the reference velocity and forces on each of them;

3. **actuatorForceLineBaseClass** class finds the intersections of each blade with the mesh and applies the force on them always following EVM, it refers to the HAWTbladeLine class to update the line position;

4. **actuatorForceBaseClass** is the lowest level one, the one which inserts the volume force on the domain and evaluates it according to the calculation of the aerodynamic forces applied on each actuator point by means of EVM approach, then divides the vectorial sum of lift and drag force (convoluted by the regularization kernel) for the volume in which the point is located. This procedure is performed at every time step and the force calculation is then inherited from upper level class HAWTapplyLineForce, from which, in turn, HAWTbladeLine extrapolated the outputs and gives them to the HAWTurbine class.

In synthesis highest level classes interface with the case set-up utilities and dictionaries from which they characterize turbine nominal properties and operative conditions, which are in turn sent to lower degree classes to find interpolation with the mesh and calculate the forces in the single points. On their side the computation methods of lower classes are inherited from the bottom to the top for force calculation on larger scale. For the complete list of the libraries see the dedicated Appendix.

4.4 Code Implementation

The crossed numerical and experimental extensive validation campaigns performed at the POLIMI in the previous Thesis Work enlighted the robustness of the code

also, at first sight, for the analysis of unsteady problems. In the past Bernini already implemented in the code the possibility of moving turbine with prescribed surge motion law. In next experimental campaign at POLIMI, for OC6 program, some tests of platform pitch turbine effects on aerodynamics are scheduled, and this revealed the possibility of comparing the wind tunnel test outputs with numerical simulation ones, in order to further validate the AL code.
One of the main scope of this Thesis Work was indeed the implementation of the turbine platform prescribed motion around a pitching axis. First of all it is useful to write the pitching equation of motion of the turbine hub, which is described as dependant on a mono-harmonic oscillation:

$$\theta_P(t) = A_P \sin(2\pi f_P t) \tag{4.6}$$

where θ_P represents the angle of oscillation with A_P as its maximum amplitude (in degrees) reached by the pitch rotation, which varies in time according to a certain frequency f_P (in Hertz). Time derivative of this quantity has to be defined, since its calculation was needed when calculating the velocity of pitch displacement.

$$\frac{d\theta_P}{dt} = 2\pi f_P A_P \cos(2\pi f_P t) \tag{4.7}$$

Given these two quantities the turbine hub equation of motion can be easily derived:

$$\overrightarrow{x}_{HUB}(t) = \overrightarrow{x}_{HUB_0} + H_P \sin(\theta_P(t))\hat{\mathbf{i}} + H_P[\cos(\theta_P(t)) - 1]\hat{\mathbf{k}} \tag{4.8}$$

$$\overrightarrow{v}_{HUB}(t) = H_P \cos(\theta_P(t))\frac{d\theta_P}{dt}\hat{\mathbf{i}} - H_P \sin(\theta_P(t))\frac{d\theta_P}{dt}\hat{\mathbf{k}} \tag{4.9}$$

Where H_P represents the hub distance from pitch axis. Substituting 4.6 and 4.7 expressions into 4.8 and 4.9 equations the complete prescribed platform pitch rotation can be obtained.
Contrarily to surge oscillations, which equation 4.10 is described below, pitch motion interests multiple degrees of freedom of the hub motion causing the skewing of the inflow with respect to the rotor and hence different loads on the blades according to their position with respect the turbine hub: if the blade is above it during a leeward displacement loads are expected to reduce with respect to the mean rotor, while in windward motion they should increase; the opposite happen if the blade is below the hub.

$$\overrightarrow{x}_S(t) = A_S \sin(2\pi f_S t)\hat{\mathbf{i}} \tag{4.10}$$

The code was hence able to simulate the turbine harmonic motion both in surging and pitching giving the possibility to extract and compare the performances and wake generation of the turbine under this two diffent motion laws, with the aim of going deeper in the knowledge of unsteady effects on wind turbine aerodynamics and characterizing distinctly the effects of one degree of freedom with respect to the other ones and validating the predictive capacity of this numerical tool.
As one can guess from the previous section, the motion implementation was written at highest level class (i.e. **HAWTurbine**). Following the path of code modification

performed for the surge case, new position and velocity equations were written in the dictionary which updates the turbine hub position and consequently all the associated blades, actuator lines and points in accordance to the imposed motion, namely *HAWTRotateTurbine*. Since the work was implemented in C++ the introduced variables needed a declaration and reading steps, but since they are mere coding matter they were reported just in the Appendix related to code implementations. More relevance may have instead the code lines implemented in the aforementioned dictionary *HAWTRotateTurbine*:

```
PitchAng=PitchAmp*sin(2*Foam::constant::mathematical::pi*
         PitchFreq*runTime_.value());


ddtPitchAng=2*Foam::constant::mathematical::pi*
            PitchFreq*PitchAmp*
            cos(2*Foam::constant::mathematical::pi*PitchFreq*
            runTime_.value());


hubVel=H*cos(PitchAng)*ddtPitchAng*SurgeDir-H*sin(PitchAng)*
       ddtPitchAng*PitchDir; //platformPitchVelocity


hubPosition=hubPositionStart+
            H*sin(PitchAng)*SurgeDir+
            H*(cos(PitchAng)-1)*PitchDir;//platformPitchPosition
```

This part of the dictionary shows how the codes makes turbine motion possible once assigned just the values of PitchAmp, PitchFreq and H, corresponding respectively to A_P,f_P and H_P parameters of the previous equations.

Chapter 5

Simulations set-up

In this chapter all the choices made to build the numerical outputs are presented, it mainly focuses on simulation parameters selection and the reasoning behind this whole set-up, whose aim is to reproduce in the most accurate way the wind tunnel experimental results.

5.1 Domain choice and discretization

Even though the actual dimensions of the GVPM (i.e. POLIMI wind tunnel) are 13.84 m in width(y-direction), 3.84 m in height (z-direction) and 35 m in length (x-direction or wind direction), the simulation domain chosen features different measures in vertical and free-stream wind directions for the following reasons: according to measures and estimations the BL thickness is 0.125m for the vertical wind profile, thus the height of the domain was fixed to 3.59m (since BL thickness was subtracted both at top and bottom walls) and at the top and bottom the a slip boundary condition was assigned (in this way the BL meshing task, which is considered difficult and may cause instability for a LES simulation, was avoided). Regarding the axial extension, it was decided to enlarge it in order to avoid the presence of the rotor induction at the inlet and to go in a condition approaching the far-wake one: hence given the rotor diameter D equal to 2.38m the inlet section was placed 5D upstream the turbine and the outlet one 15D downstream. The adopted computation domain was thus a simple 13.84m wide, 3.59m high and 47.6m long parallelpiped and the turbine hub was placed in point $(0, 0, 1.961)m$.
For meshing purposes the background mesh (created with *blockMesh* utility) presented a simple grading in order to start lowering the mesh size in the central region, where turbine operations have higher impact (induction and wake generation), taking care of having a smooth transition toward lateral walls, as can be seen in figure 5.6.

The *blockMeshDict* was written in a parametric way with respect to the cell dimension in the turbine rotor zone, in this way when performing a mesh sensitivity analysis the main quality parameters remain similar and acceptable. The background mesh was made only of hexahedral cells, without non-orthogonality, and it was useful to refine it locally, in order to capture accurately the wake behind the machine, the turbine wake induction and the vortexes generated at the tip of turbine blades, moreover it was beneficial since it increased the number of actuator line points for each

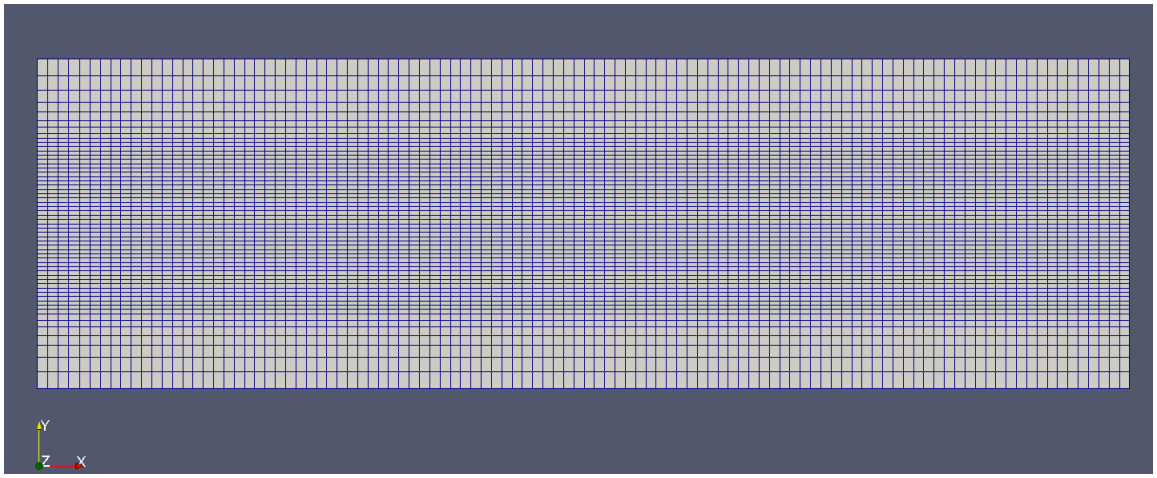


Figure 5.1. Visualization of the background mesh plus absolute domain reference axes

blade (a crucial parameter to improve the simulation fidelity, in fact good practice requires to have at least 30/40 points per blade). The refinement was obtained by means of *refineMesh* utility: this, in combination of a *topoSet* utility, halved the cells edges dimensions in the region specified by the *topoSet* dictionary. This procedure was performed twice: first in a cylinder starting 1D upstream the rotor and extruded along the wind direction till the end of the domain with a radius twice the diameter of the rotor; then inside this zone a further refinement was applied along an other cylinder with a radius equal 1.4 rotor radius, extruded symmetrically 0.13D upstream and downstream the rotor. This routine aimed to capture the aforementioned flow and wake features by keeping an acceptable mesh quality. The region of extrusion

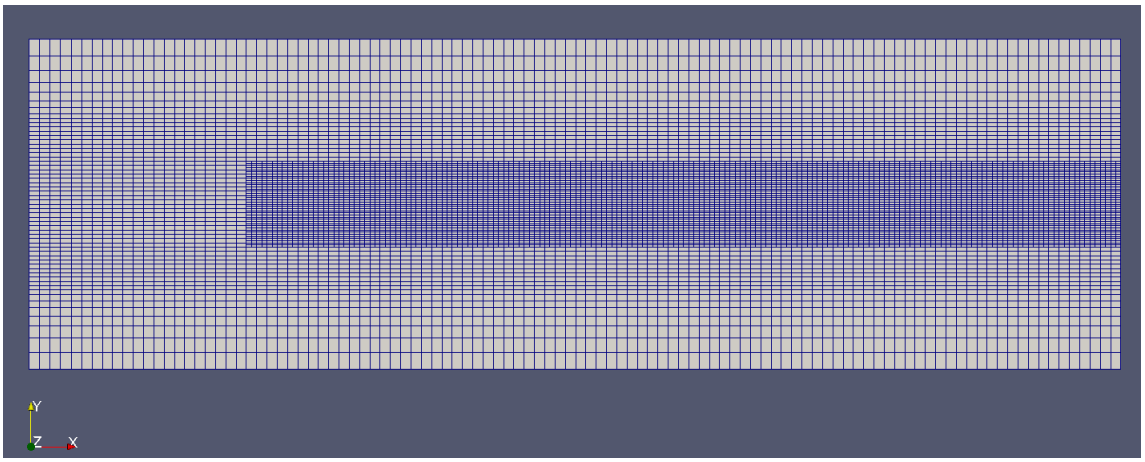


Figure 5.2. Visualization of the mesh with the addition of wake refinement zone

of the inner cylinder extension was chosen in order to guarantee that the rotor, the EVM sampling line and the tip blade vortexes always belong to this region also under unsteady operations.

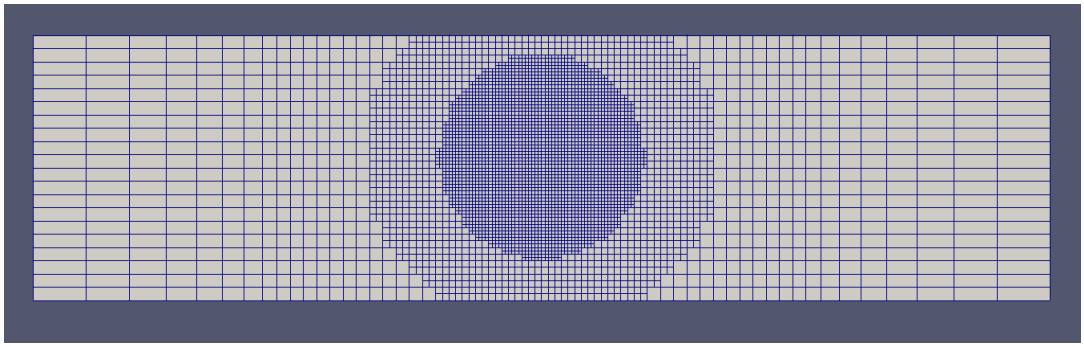


Figure 5.3. Clip visualization to show the inner rotor cylinder refinement zone

5.2 Turbulence Model Selection

Thanks to the turbine absence and the consequent hexahedral mesh the AL codes are often suitable for LES simulations and also in this work it was decided to apply them, thus giving the possibility of simulating large scale turbulence. When dealing with a LES simulation, in fact, turbulent structures with a dimension above a certain filter(namely Δ) are directly solved while smaller ones are modelled by means of a SGS) model. The filtering procedure is performed by the convolution of a generic field with a filter kernel function (which is associated to a characteristic time (simulation time-step) and length (Δ)). In this problem the filter size cube root approach was chosen, in which $\Delta = l_{cell}$. Once filtering is applied the the flow field quantities (p,$\mathbf{U}$) can be seen as the sum of filtered ($\overline{q}$) and sub-filtered (q') parts and this decomposition introduced in the original Navier-Stokes equations generates additional unknowns located in the *SGS stress tensor*, which here was modelled with Smagorinsky model. The resulting filtered momentum equation considering AL volume force is presented below, 5.3:

$$\mathbf{U} = \overline{\mathbf{U}} + \mathbf{U}' \tag{5.1}$$

$$p = \overline{p} + p' \tag{5.2}$$

$$\frac{\partial(\rho\overline{\mathbf{U}})}{\partial t} + \vec{\nabla}\cdot(\rho\overline{\mathbf{U}}\overline{\mathbf{U}}) + 2(\nu + \nu_{SGS})(\nabla\overline{\mathbf{U}} + \nabla(\overline{\mathbf{U}})^T) + \mathbf{f}(x,t) = -\nabla p \tag{5.3}$$

with cube root approach (C_S=0.1678):

$$\nu_{SGS} = (C_S\Delta)^2(\nabla\overline{\mathbf{U}} + \nabla(\overline{\mathbf{U}})^T) \tag{5.4}$$

5.3 Constant

The implementation of AL in OpenFOAM made necessary the creation of new dictionaries in the Constant folder of the simulation case, more in detail a dictionary called *turbineProperties*, another one called *actuatorForceParam* and a folder, *airfoilProperties*. The scope of these additional files is to assign the properties of the simulated

operating turbine and the set-up parameters of the AL code. In the *turbineProperties* dictionary rotor hub position, inner and outer radius, rotational speed, yaw, blade pitch (to feather) values are given. Also here platoform pitch and surge amplitude and frequency are assigned, for the pitch additional distance from pitch axis term is imposed. Last airfoil section properties (chord, twist angle, span position) are assigned. The polars of these airfoils are all written in the *airfoilProperties* folder, in which there is one polars file for each airfoil. The parameters of the AL set-up are then assigned in the *actuatorForceParam* file: the epsilon (relative to cell dimension in which force is applied) of the regularization kernel, EVM upstream distance and length of the sampling line values are assigned, all in relative terms with respect to the cell dimension. For this Thesis work the Turbine assigned values were coincident with the ones of the turbine model described in Chapter 3 for the OC6 experimental campaign, as showed in the Table below.

Load Case	$R_{out}[m]$	$R_{in}[m]$	Ω [rpm]	Blade Pitch [°]	Hub Position
1.1	1.1906	0.089	240	0	(0,0,1.960)
1.2	1.1906	0.089	265	12	(0,0,1.960)

Table 5.1. *turbineProperties* set-up for fixed turbine cases

The airfoil polars were imported from the OC6 database, for the Reynolds regime at which the turbine section was supposed to work (figure 3.4).
Last consideration on constants set-up as to be done on the actuator force parameters (*actuatorForceParam*): the tuning of EVM performed by Bernini Caccialanza ([38]) recommended to set the upstream distance and sampling line length respectively to 1.5 and 5 times the cell dimension (d). In their work was also stated that when dealing with EVM the epsilon on dimension (ϵ/d) value could be kept at 1 without causing wiggles, while in other AL codes, like the SOWFA, this term should be at least equal to 2 to avoid non physical oscillations on the solution. Hence the EVM parameters were set as advised by Bernini Caccialanza while some comparisons (only for RATED case) were performed with (ϵ/d) equal both to 1 and 2, to see whether discrepancies were large in the two cases. As will be shown in the following Sections thrust,torque and power outputs highlighted an underestimation of them in case of epsilon equal to 2, in particular while performing the mesh sensitivity analysis it continuously decreased with when increasing the cells number; hence for the the other simulations this parameter was assigned to 1.

5.4 Boundary Conditions

The aforementioned domain choices combined with the adoption of LES models are strictly related to the boundary conditions (BCs) choices for the simulations. In fact, as stated before, at the inlet boundary it was possible to consider completely undisturbed conditions for the wind (no wind turbine induction was detected there), hence the velocity imposed was equal for all the boundary to the free-stream condition (V_0) according to the considered load case (namely 4 m/s for RATED and 6 m/s for ABOVE ones), the operational points at which the turbine performances were studied

in this work. According to the transverse measures obtained in the GVPM the wind profile entering the turbine could be considered homogeneous and with Turbulence Intensity (TI) at the 2%. For what concerns lateral walls *slip* condition was imposed: this is based on the consideration that the lateral blockage in the wind tunnel had low impact, moreover in this way the BL meshing was avoided allowing a stable resolution also with LES. Then at top and bottom walls the same condition was applied, this time the choice can be seen as more consistent, according to the considerations made in the domain choice and discretization considerations. Last, at the outlet the *inletOutlet* condition was imposed for a better stability than the simple *zeroGradient* one, avoiding backflow effects. For what regards the pressure, since the solver used was non compressible the values had to be expressed in relative terms with respect to the ambient pressure (whose value is 0 Pa). The inlet and sidewalls condition adopted was *zeroGradient*, while at the outlet, full pressure recovery was deemed by imposing a fixed value equal to zero. Concerning ν_{SGS}, it was chosen to not account for turbulence at the inlet. It is be possible to introduce it, but this should require precursor simulations, and thus it was considered too expensive in terms of computational costs for this thesis work, also because turbulence can be considered as an high frequency disturbance which should not affect too much the pitch and/or surge oscillations effect.

Wall	p	**U**	ν_{SGS}
Inlet	*zeroGradient*	*fixedValue* (V_0,0,0)	*fixedValue*(0)
SideWalls	*zeroGradient*	*slip*	*slip*
Outlet	*fixedValue* (0)	*inletOutlet*	*zeroGradient*

Table 5.2. *Boundary Conditions set-up*

5.5 System

Trusting Mancini's Thesis agreement among numerical and experimental results, it was chosen to keep the same numerical scheme and solutions set-up. More in detail for what concerns the numerical schemes for the equation discretization:

- *Time Scheme*: **Crank-Nicholson 0.9**, a hybrid I-II order scheme with high blending coefficient (0.9), that makes it more similar to a II order. The short time step applied made although its influence low;
- *Gradient Scheme*: **Linear**, purely second order scheme based on central difference;
- *Divergence Scheme*: **QUICK**, III order scheme for the convective term to have high accuracy, in particular for the description of tip vortexes convection;
- *Laplacian Scheme*: **Linear corrected**, II order scheme with the correction for non orthogonal cells. Actually mesh orthogonality was always good (average below 5°), thanks to mesh configuration, but this did not cause instability issues.

As already mentioned in the AL chapter description the solver used was based on PISO algorithm logic, with the introduction of the AL force at each time step. This solver is suitable for the resolution of non-compressible and unsteady flow problems and grounds on one big hypothesis: if the time-step is small enough the convective term of the momentum equation can be neglected, this means that the flow motion is governed only by the pressure gradient term. Hence the algorithm has to solve the pressure equation (made of two steps, namely momentum predictor and pressure correction) as many times as imposed by the number of correctors at each time step. Thanks to the hypothesis, at each time step the convection is negligible and thus momentum predictor does not change during the loop, it us updated only the following solution instant. Of course PISO is recommended only when its supporting hypothesis holds, hence it is of crucial importance having a highly resolved problem in terms of time-step. This restriction matched the request of the small time step resolution of the LES simulations, and thus PISO loop fitted well for this purpose. The number of internal correctors was fixed to 3, as commonly accepted best-practices. Furthermore the PISO explicit logic was perfect for the implementation of EVM, thus its introduction did not cost too much in terms of resolution time.
Regarding the solution of linear equations a GAMG (Generalised Algebraic Multi-Grid), with a Gauss-Seidel smoother, method was used for the calculation of p. The logic of GAMG method is that a coarser mesh is more capable to eliminate high-frequency errors in the calculation than a finer one and since a low frequency error for a small cell becomes an high-frequency one for a bigger cell, the solver continuously consider a coarser mesh during the resolution. Furthermore the information included in the finer cells is consequently contained in a coarser one, there is no need of recreating multiple meshes for this purpose. For u and ν_{SGS} a precondtioned bi-conjugate gradient (PBiGC) method was used, with a diagonal incomplete lower upper (DILU) preconditioner, suitable for asymmetric matrix. The tolerance level for each quantity was set to $10^{-}6$ for the final iteration and the residual behaviour of every simulation was checked to verify proper convergence of the solution.

5.5.1 Time Step Definition

As should now be clear, the selection of the simulation time-step (assigned in the *controlDict*)is of paramount importance to obtain consistent outputs from the simulation and several considerations had to be done in this thesis work in order to achieve proper time steps. More in detail these are the criteria exploited for the choice:

1. best practices suggested to keep LES simulation **Courant Number below 0.5**, in this way the need of solving the grid scale turbulent structure was guaranteed;

2. AL simulations proved to be reliable only if **the blade tip was not crossing more than one cell per time-step**, in order to maintain a smooth application of the source term;

3. the Δt had to generate a sampling frequency of the problem ($1/\Delta T$) which was an integer multiple of the platform pitch/surge one of the unsteady simulation to **avoid the leakage effect**

Among the three constraints the most restrictive was the second one, in fact in all the run simulations, once guaranteed this condition the maximum Courant number was around 0.2 and the average value even one order of magnitude smaller, and the leakage limitation acted just to round (for defect) the time-step imposed by the tip limitation. Lastly simulation were run with constant time-step to facilitate the post processing operations.

5.6 Simulation Choice criteria

Since the objective of this Thesis Work was the comparison between platform pitch and surge motion effects in terms of aerodynamic performances and wake generation there emerged the necessity of finding a parameter which allowed to state that a pitch and surge condition were "similar". In fact it has to be remarked that surge motions is linear while pitch is a rotation which causes skewed load on the turbine rotor. As already introduced, an experimental campaign is scheduled in the next months of the current year here at POLIMI in order to measure the effects of platform pitch and a test matrix with several cases, namely Load Cases, was already delivered, but since there are no outputs yet not all the indicated cases were run. This is another reason why a criterion for the Load Case (LC) choice arose. Among the unsteady parameters listed in the Introductory Chapter the $\Delta V/V_0$ one seemed to be most powerful to determine a priori if a pitch and surge simulation are similar, in fact reduced frequency analysis would require the knowledge of the wind velocity at each span of the three blades whitin every time step and does not take into account the amplitude of the excitation, while the wake reduced velocity implies the knowing of wake velocity and these aspects made them difficult to be used in pre-processing phase as term of comparison. Another advantage of the $\Delta V/V_0$ value is explicitly dependent on the oscillating disturbance frequency and amplitude and on the main input that influences performances output magnitude, V_0: this relation also is formulated in a very synthetic and direct way. For surge and pitch cases ΔV expressions are presented below:

$$\Delta V_S = 2\pi A_S f_s \tag{5.5}$$

$$\Delta V_P = 2\pi A_p f_p H_p \tag{5.6}$$

Grounding on this criterion, among all the possible pitch simulations the following ones were chosen in order to compare them with some of the surge load cases listed in the OC6 definition document.

Surge LC	A_S [m]	f_S [Hz]	$\Delta V/V_0$ [-]
2.5	0.035	1	0.055
2.6	0.01	1	0.016
2.7	0.008	2	0.025
2.8	0.004	2	0.013

Table 5.3. Surge Load Cases definition

Pitch LC	A_P [°]	f_P [Hz]	$\Delta V/V_0$ [-]
p2.5	1.4	1	0.062
p2.6	0.4	1	0.018
p2.7	0.6	2	0.027
p2.8	0.3	2	0.014

Table 5.4. Pitch Load Cases definition

It is worth noting that each surge simulation was compared with its pitch counterpart, according to the aforementioned parameter, taking also care that the frequency of the oscillation was the same for homologue cases.

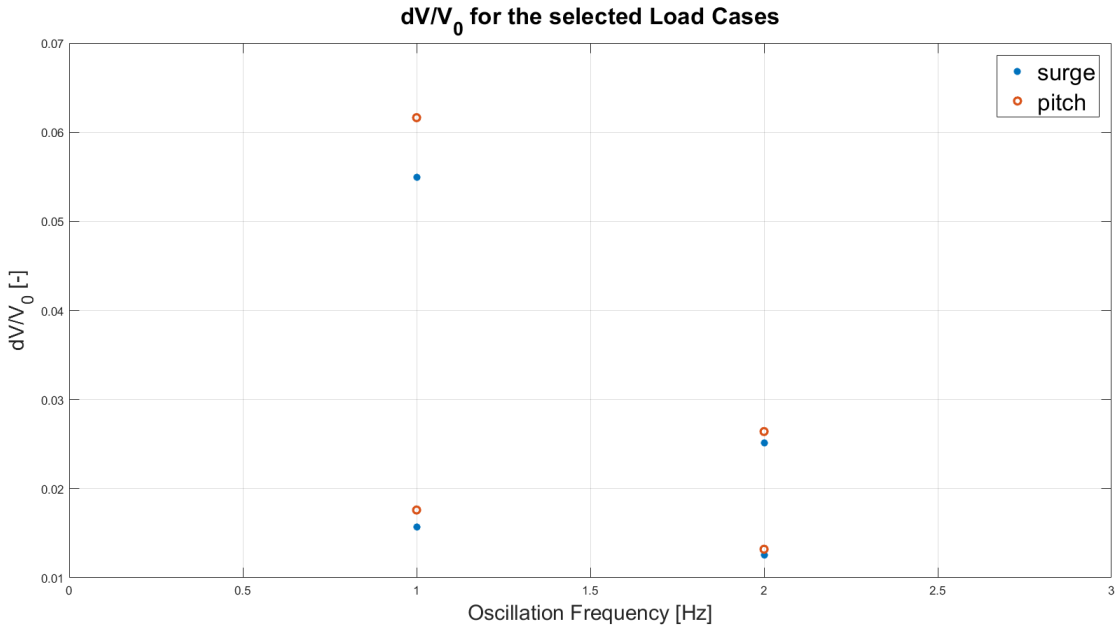


Figure 5.4. Pitch and Surge studied cases according to $\Delta V/V_0$ parameter

In the figure 5.6 it is easier to appreciate the similarity in terms of $\Delta V/V_0$ of each surge load case with its pitch counterpart.

5.7 Probes placement

5.7.1 Azimuthal Averaging Technique (AAT)

There exist in literature examples of techniques for the extraction of the wind turbine velocity induction both for experimental and CFD studies (in this article a comparsion of these methods is given, [40]). The original scope of this applications was to accurate reproduce the angle of attack to obtain 3D corrected high-fidelity polars. It has to be noted that all these methods are developed only for fixed turbines, one of the only attempts of induction sampling under unsteady conditions was performed by Mancini in his Thesis Work, even if it was the extension of steady technique application. The approach he used is called Azimuthal Averaging Technique, AAT, (Mikkelsen, [41]), and was chosen due its simple implementation. The AAT consists in the extraction of the velocity at the disk by means of an averaging procedure between an upstream and downstream sampled velocity. The velocity has to be sampled on two planes of probes laying respectively a little bit ahead and behind the rotor. The probes planes consist of one third of circumference arcs at different span fraction, with a significant amount of probes for each curve. Then for every time step the sampled velocity at each span fraction is averaged, both upstream and downstream, and finally on these averaged outputs the mean is computed, generating the desired velocity at the disk. The consistency of this method bases on the proper quantification of that aforementioned *"a little bit ahead and behind the rotor"*: the method is valid under 2D flow assumption, so the probes cannot be too close to the rotor, otherwise 3D effects will impact too much the calculation, anyway they probes have to sample the velocity ideally on the same streamline and this forces to place them not too far. The trade-off distance selected by Mancini was thus 0.1D upstream and downstream the turbine. The bidimensionality required the method does not hold anymore where the flow features relevant vortexes, namely at tip and hub of each blade. Thus in this work it was decided to place sample lines from 0.4 to 0.8 span fraction (with 0.1R step), since in previous the previous work at 0.3R the vortexes presence was still relevant. Another limitation of this technique is that requests axial inflow, a condition that is not satisfied during platform pitch operations. It was decided to introduce this study because it could be interesting to compare the velocity measured with this approach with the EVM sample velocity to see induction value both in steady rotating turbine case and in surge ones, since AAT proved to give also in surge applications. To check the validity limits of this technique a quantity was introduced, which is the ratio between two dimensional velocity norm and tridimensional one, this value reduction below a certain threshold means that 2D flow assumption does not hold anymore and thus AAT too.

$$\overline{U}_i^{upstream} = \frac{\sum_{j=1}^{20} U_{ij}^{upstream}}{20} \tag{5.7}$$

$$\overline{U}_i^{downstream} = \frac{\sum_{j=1}^{20} U_{ij}^{downstream}}{20} \tag{5.8}$$

$$\overline{U}_i^{D} = \frac{\overline{U}_i^{upstream} + \overline{U}_i^{downstream}}{2} \tag{5.9}$$

$$a_i^D = 1 - \frac{\overline{U}_i^D}{V_0} \tag{5.10}$$

where i-subscript states for the blade span position with respect to blade radius ($i = 0.4, 0.50.60.70.8$). Note same calculations can also be performed for the tangential component of the velocity, but for the thesis purpose it resulted more relevant to analyse the axial one since loads variation of the the two investigated platform motions affects basically flow velocity in this direction (actually the pitch influences the flow also in vertical direction, but the studied load cases featured very low oscillation amplitudes leading to a gross axial inflow aproximation). Among all the possible induction extraction techniques there exists one, prorposed by Herráez, which is very interesting and straightforward ([42]): it consists in the application of a PIV line along the bisector of two wind turbine blades since along that line, the flow symmetry causes the downwash of the blade ahead of the bisectrix to counterbalance the upwash of the blade behind it. Hence, the velocities extracted from the rotor plane are just influenced by the wake induction and do not need to be corrected for blade induction influences (this fact is analitically demonstrated in Herràez article).

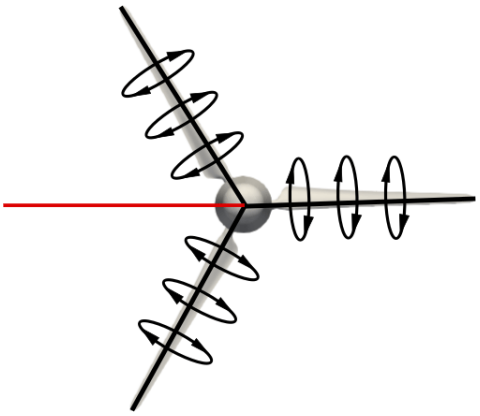


Figure 5.5. Schematic of the two consecutive blades counterbalance of downwash and upwash

An attempt to reproduce this measurements set-up was made during this thesis work but the Author did not succed in implementing it in the code. Despite the easiness of this concept it showed difficult to write it in the AL code. The general idea of the procedure is here written to give an input for the implementation in possible future works: an additional actuator line has to be introduced in the code (or even three if one would like to average the results of the bisectors), able to move with all the actual blades, then velocity sampling have to be performed with a local approach but the forces application has to be avoided (and this was the main factor that avoided the implementation success). Starting from this preliminary instructions one might try the addition of this technique in the code to compare it with AAT. Of course this approach is valid only for axial inflow, but given the same reasoning made for AAT also for low amplitude pitch oscillations might give reliable results.

5.7.2 Wake Measurements

The definition document of OC6 described the set-up required to measure flow velocity in the wake, as reported in Chapter 3. To made possible a future comparison with experimental data and validate the numerical model, several probes were inserted in the computation domain in the same locations indicated in the document. By the way for this Thesis Work further probe lines were placed, in fact it was considered useful to compare wake measurements not only with an horizontal transverse line but also with a vertical one since pitch motion skewed operation might result in a non-symmetric flow distribution along that direction. This probes cross pattern was placed 2.5D downstream the turbine, in the near wake zone, and was repeated 5D downstream with the aim of appreciate the wake evolution behind the turbine.

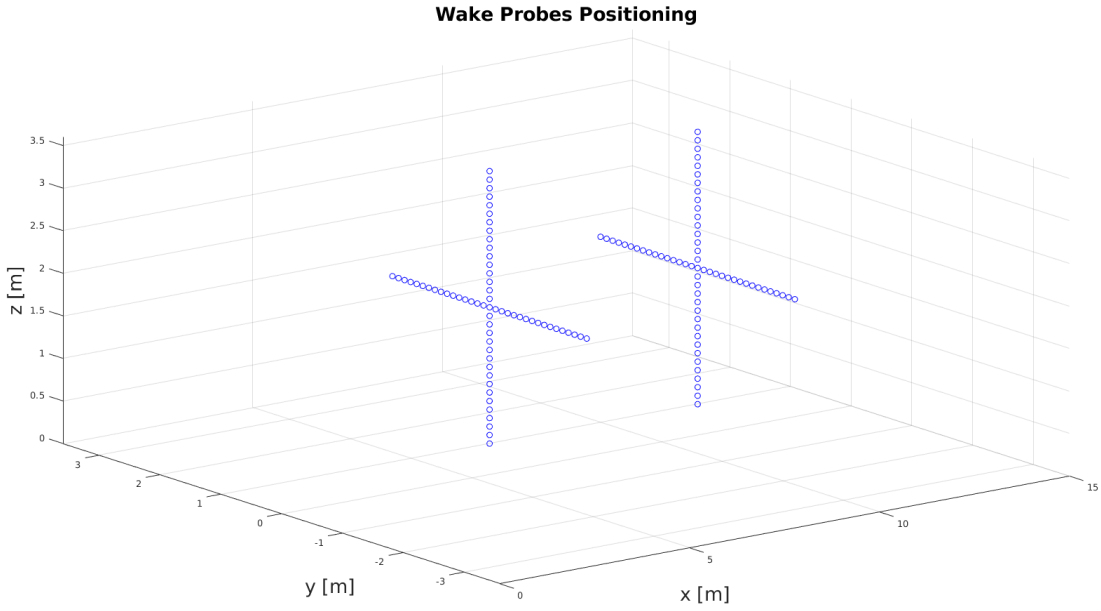


Figure 5.6. Probes placements for wake measurements

5.8 Mesh Sensitivity Analysis

5.8.1 Meshes lists and set-up

As already mentioned a mesh sensitivity analysis was performed for this Thesis work. It was made only for the RATED condition load case, with fixed turbine. the choice of developing this study just in one single case is justified by the fact that the domain size was already significantly enlarged in along wind direction minimizing the effects of wake induction at the inlet and moving toward far wake condition at the outlet, hence it was supposed that one oscillation with an amplitude of the order of the centimeters would not affect too much the flow conditions on the domain boundaries. The simulations were carried out thus just for the L.C 1.1 for different cells number and the comparison parameters selected for the analysis were the mean turbine Thrust

and Torque giving always an eye to the minumum number of actuator line points per blade (which has to be at least equal 30/40 to have reliable results) and on the simulation time step according to all aforementioned constraints. Moreover, in order to verify the effects of the smearing parameter ϵ on AL code with EVM, the analysis was performed both with ϵ/d equal to 1 (as advised by Bernini Caccialanza [38] and 2 (the minumum value recommended by other authors. like Churchfield [43]).

MeshID	#Cells [millions]	#ALpoints	Δt [s]
M#1	0.4	30	0.002
M#2	3.3	50	0.001
M#3	11.2	75	0.0005
M#4	21.6	97	0.0004

Table 5.5. Meshes main characteristics

5.8.2 Mesh sensitivity outputs and selection

Before entering in the results review it is worth to make this preamble: when performing a CFD simulation it is recommended to perform the mesh sensitivity analysis to reach the mesh grid independence. One can confirm to have achieved this target once a selected parameter does not change anymore with the increase in mesh refinement. Anyway in this work there was a limiting factor which avoid this task completion, the computational resources and capacity. In fact simulation were run with the POLIMI HPC (*CFDHUB*)machines which cannot always provide an extremely large number of cores to each user. Hence it was chosen a mesh according to a trade-off between computational burden for the cluster facilities and accuracy in the results, which was quantified in terms of coherency with the already available experimental data. The wind tunnel performances and wake outputs of RATED (LC1.1) case were present and thus were taken as reference.
As can be seen from figure 5.7, both Torque and Thrust values were still sensible to mesh refinement, leading to conclusion that the independence from the cell dimension was not achieved yet. Anyway both in $/d$ equal to 1 and 2 the relative variation of the studied quantities kept lowering with an increasing number of cells (in particular in the case equal to 1). It is clearly visible instead that if the smearing function equals the value of 2 the results strongly underestimate the performances. Hence this confirmed the recommendations of Bernini Caccialanza for $/d$ equal to 1 in case of EVM adoption in the AL model.
Coming back to mesh selection: once established the relative reduction in terms of performances output and taking care that a reasonable computation time (about an hour for one second of simulation) with the 21.6 millions cells required 100 cores (with a total amount of 320 cores in the cluster), first it was chosen to not further investigate the sensitivity to the mesh and last to adopt M#3 to run all the remaining load cases. Simulations were run on 40 processors to avoid a too busy cluster, and the time implied to run a second of simulation was slightly higher than one hour when the nodes were not too loaded.

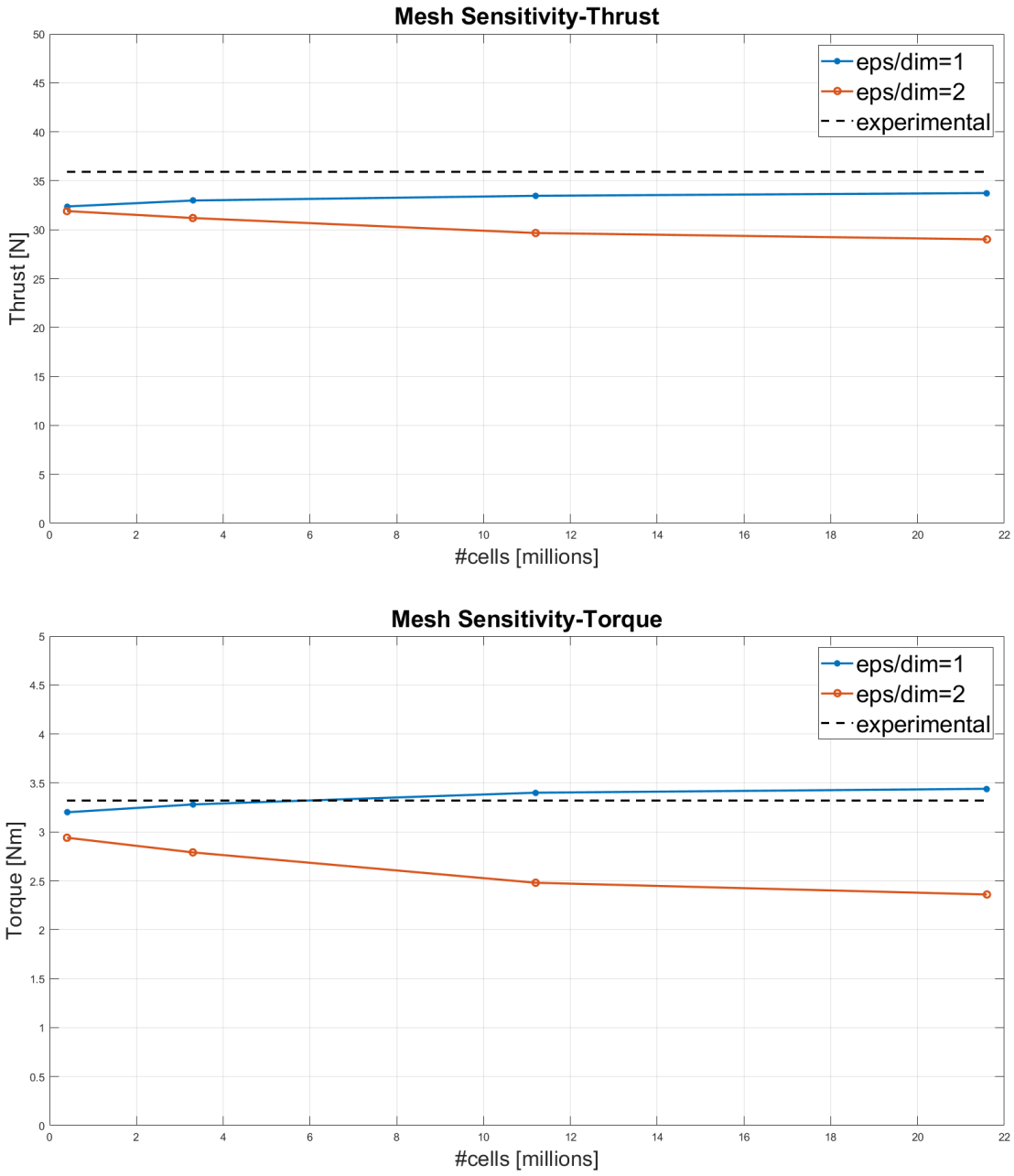


Figure 5.7. Mesh sensitivity analysis outputs for Thrust and Torque

Chapter 6

Comparison of simulations outputs

In this chapter the simulations outputs are reported, with a deepening on the comparison among pitch, surge and, when available, experimental data and reasoning on their analogies and discrepancies. Only the most relevant results are reported in the chapter, while for the complete list of figure make reference to related Appendix.

6.1 Fixed turbines output

First of all it is important to show the two fixed turbine cases results (namely LC1.1 and LC1.2), which were taken as a first tool to validate the reliability and accuracy of the code comparing its outcomes with the already available wind tunnel outputs. Furthermore the RATED case initial velocity boundary condition ($V_0 = 4m/s$) coincided with the ones of all the pitch and surge analysed load cases, thus it was considered as their reference case for the comparison of the main averaged outputs like integral Thrust and Torque, local blades forcing and wake profile.

6.1.1 Integral Performances

First analyzed outputs were integral Thrust and Torque (and thus also power, since it is just Torque values times the turbine rotational speed, Ω).

Load Case	AL Thrust [N]	Exp Thrust [N]	AL Torque [Nm]	Exp Torque [Nm]
RATED(1.1)	33.46	35.91	3.40	3.32
ABOVE(1.2)	20.86	18.28	3.34	3.06

Table 6.1. Integral performances of fixed turbine cases

As one can observe in the LC1.1 the thrust is underestimated while the torque is accurately. The causes of this lower value may be associated to the absence of the nacelle which avoid strong vortexes generation and recirculation even lower span contribution weights much less on the machine performances, a further factor might be the absence of a dedicated tip correction method which could reproduce more accurately the tip vortexes effects at the end of the blade span. Moreover wind turbine experimental scaled model included nacelle, turbine and effective blades reducing the

available section for the wind passage and causing thus an higher increase of velocity (for the mass conservation principle), and hence available power with respect to the simulated turbine.
Going through LC1.2 results, the Thrust and the Torque on the machine are now overestimated. Hence the causes cannot coincide with aforementioned ones for RATED case, the differences here might rather depend on the accuracy of EVM method with lower Tip Speed Ratio (TSR): as evidenced Bernini Caccialanza's thesis this technique suffers more of inaccuracy at small TSR, When the rotational speed decreases the airfoil profile is prone to stall and this phenomenon causes detachement of the flow which become more complex and the method encountering this kind of flow become less accurate since it filters some flow features by its average computation procdure.

6.1.2 Local Forces evaluation

The study of loads evolution along the blade span is here reported. For each blade the forces distribution are presented in term of ensamble average on the rotational period. In this case practically identical average loads were expected for the blade since subjected always the same inflow (indeed we were in steady conditions).
As expected, loads distribution are most equal for all the three blades of the rotor, the perceived discrepancies are present at the tip (and this should be caused by the tip vortexes generation which causes a more stochastic loading). The only relevant difference is clearly visible at the *blade0* for Rated case (figure 6.1): this can be causes by a larger tip vortex generation for that line in that operation point and may explain the thrust estimation deficit of the numerical simulation. Indeed, tangential force distribution is less scattered and in fact torque numerical tool is practically equal to experimental outcome. Radial loads are null and this fact is coherent with the linear momentum theory.

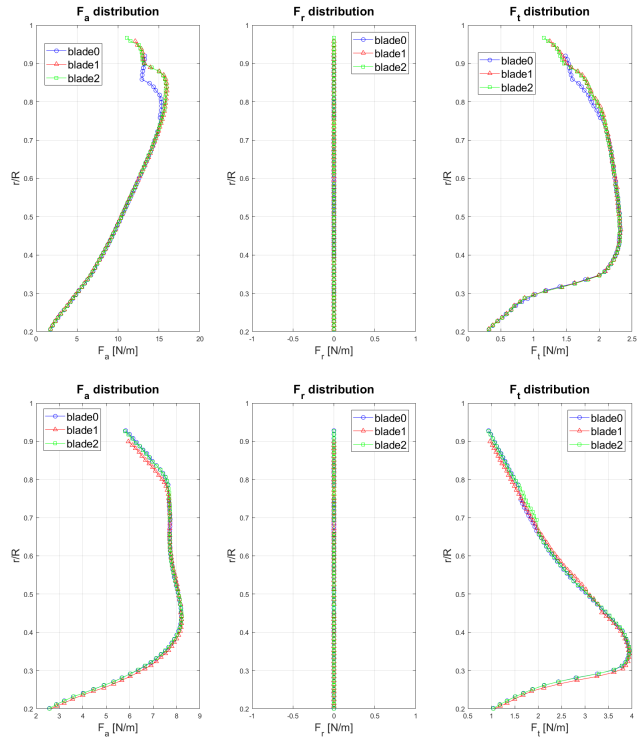


Figure 6.1. Axial,Radial and Tangential force distribution for each rotor blade, in LC1.1 (top) and LC1.2 (bottom)

6.1.3 Steady Axial induction extraction

It is then interesting to highlight the analogy of Lift coefficient trend and axial induction evaluated with aforementioned AAT. The C_L increase is associated to an high machine momentum extraction from the flow. Since the coefficient keeps getting higher with the span, one can imagine also an augmentation of the axial induction a, and this can be appreciated in the top left frame of figure 6.2 and top one in figure 6.3. The dual behaviour instead is clearly visible in the LC1.2, in which the lift coefficient reduces along the blade radius and the same does the axial induction in the evaluated considered positions. The trend of induction evaluted with AAT in LC1.1 is significant since its evolution is almost equal to the Glauert optimal distribution: RATED conditions were intended to simulate the turbine design ones. Indeed the design target is to construct a machine with highest possible performances, and these are achievable when the distribution mimics Glauert one.
All these results and plots highlighted the reliability of this numerical tool when dealing with fixed turbines, in which the linear momentum theory holds and can be considered the reference, since it describes in a accurate way the machine at low computational costs.

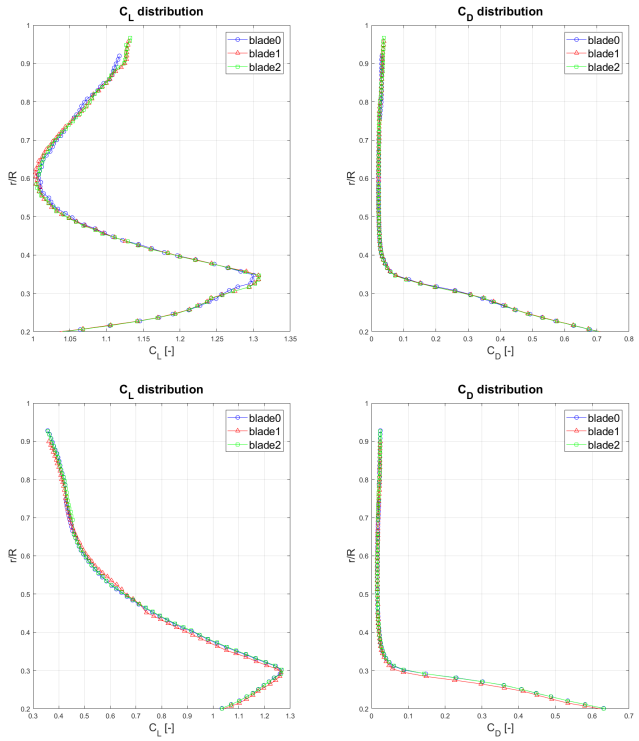


Figure 6.2. Lift and Drag distribution for each rotor blade, in LC1.1 (top) and LC1.2 (bottom)

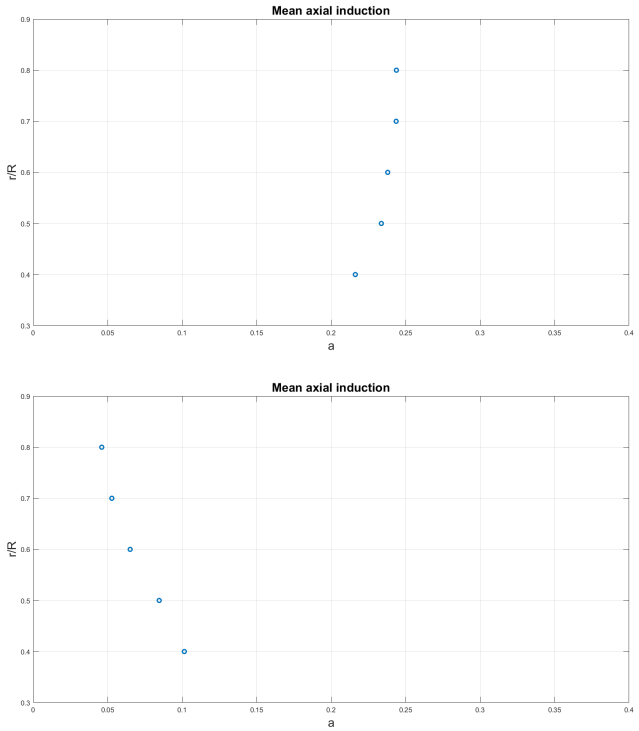


Figure 6.3. Axial induction computed with AAT in LC1.1 (top) and LC1.2 (bottom)

6.1.4 Wake Analysis

Last validation step for the numerical simulation was made by comparing the wake probes numerical output with experimental ones. Of course all the wake profiles reported in the chapter will feature one significant difference with the test outputs: the absensce of wake deficit in the center of the stream tube. in fact, as already stated, there were no nacelle and tower in the simulation. Albeit this might seems strange, the remaining part of the flow, according to the case, fits quite well the trend of experiments (anyway it would be interesting to implement the two structures in the AL codes to see how their presence alters the flow features and performances). Moreover it was also interesting to show the wake profile evolution in downstream the machine. For fixed case it was chosen to not report the vertical transverse measures since the outcome should not differ too much from horizontal one in terms of mean properties (the inflow was axial and turbine kept fixed).

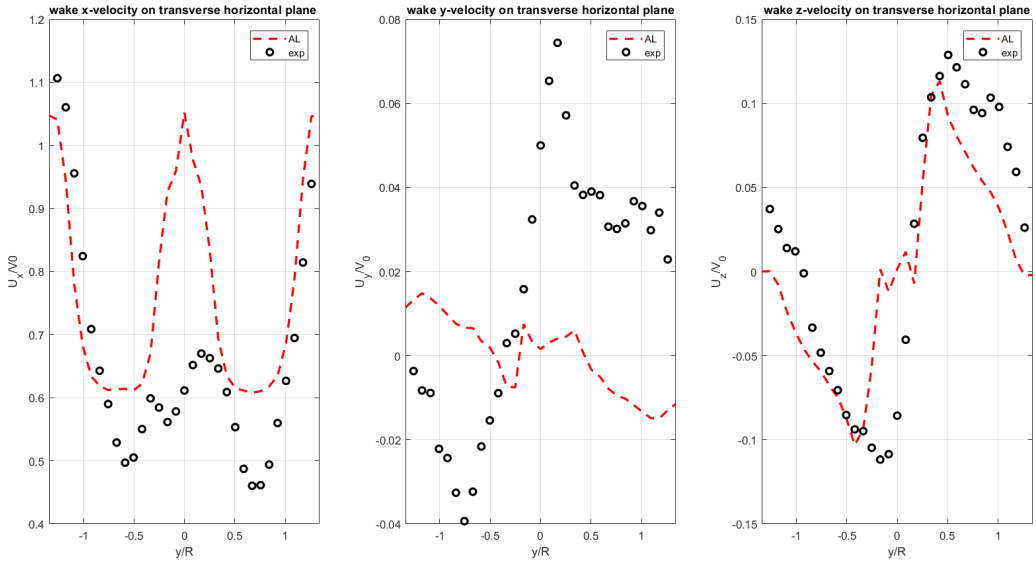


Figure 6.4. Wake numerical and experimental profiles for RATED case

The maesurements were acquired 2.3D downstream the rotor, a position that can be considered in the near-wake region, where the flow is strongly influenced mainly by the machine forces and the interaction with surroundings free stream is lower. For this reason the tower and nacelle presence is highly relevant, and can be appreciated in the first and second frame of figure 6.4: the normalized velocity with respect to inlet velocity in center of the figure is significantly different. The aforementioned obstacle effect of the two components can be also seen at the extremities of the frames: the U_i/V_0 is higher in the experimental measurements and can be explained by their presence. The two patterns, in x-direction, are similar, except for the magnitude of the deficit. The lower loss in the numerical case can be easily associated to its thrust underestimation. On the experimental profile asymmetric peaks are present, maybe due to a little misalignment, in y-direction, of the hot-wire system with respect to the turbine center. The central frame of figure 6.4 further underline tower and nacelle absence in numerical simulations, but the important aspect is that the velocities ratio

is one order of magnitude lower than for the axial one. The y-direction here coincides with the turbine radial one, and low values means that the flow does not expand too much in radial direction, in agreement with 2D velocity linear momentum assumption. Last, in z-direction, the congruence is high and this further explains the agreement of numerical and experimental torque output; this direction for the turbines represents in fact the tangential one, and the flow deviation in this direction is the one generating the moment around turbine axis.

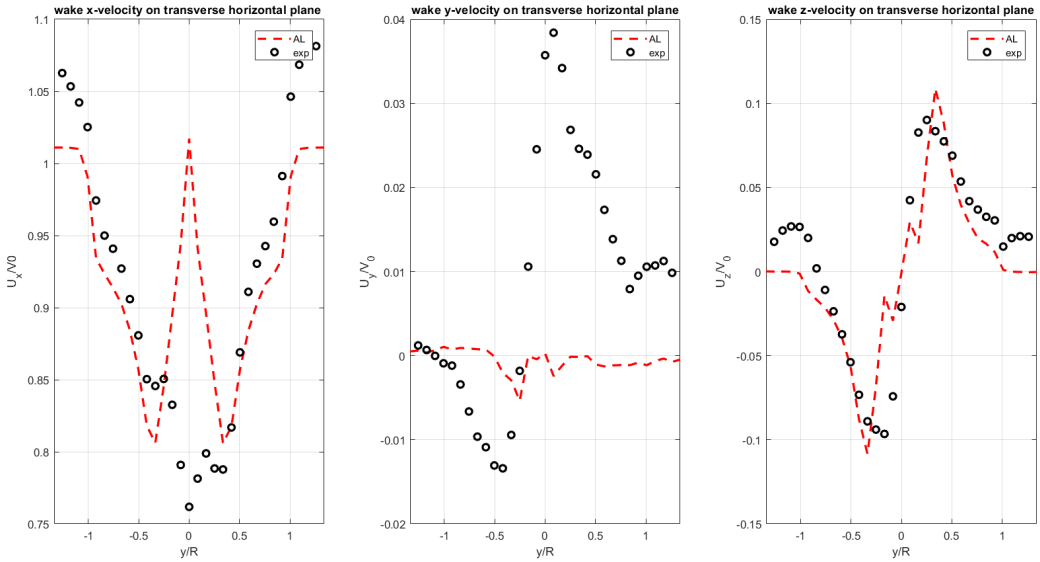


Figure 6.5. Wake numerical and experimental profiles for ABOVE case

in figure 6.5 it is remarkable the presence of wake contraction in free-stream direction in both the studies, This may be caused by the lift drop toward blade tip reducing the momentum extraction of the machine from the flow. However the difference in terms of flow velocity at the extremity is high, maybe due to the too high load reduction calculated from the numerical simulation.Same reasoning of RATED case can instead be done for the y and z velocity profiles.
Last wake comparison was instead make just to compare numerical results: the flow velocity was measured 5D downstream the turbine rotor to see how much the flow was recovering pressure and reaching far-wake condition. A significant recovery was expected to show a more compact velocity profile, less influenced by the machine interaction but rather affected by the neighbouring free-stream.

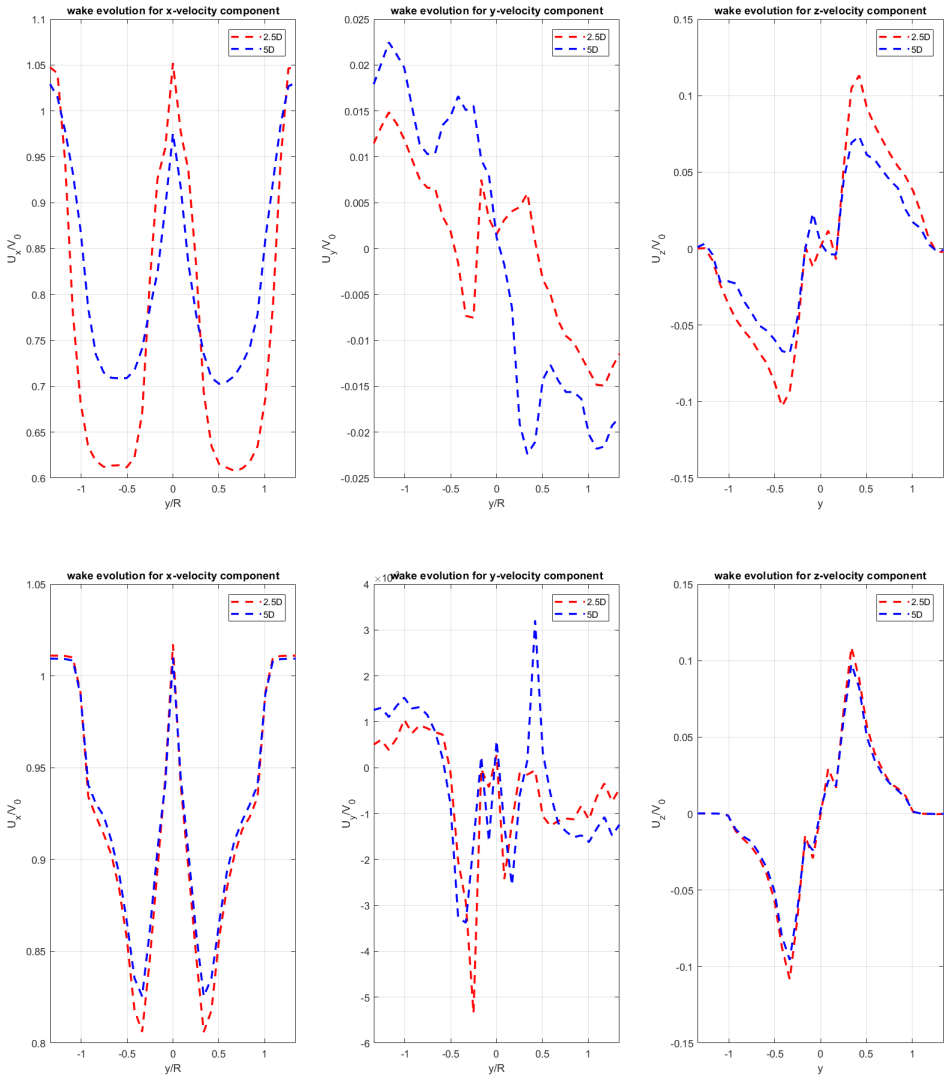


Figure 6.6. Wake evolution in flow direction for LC1.1 (top) and LC1.2 (bottom)

The most relevant velocity component is the x one, and from figure 6.6, one can notice an higher recovery in relative terms for the LC1.1 case. Anyway from an absolute point of view LC1.2 velocity profile was already recovered. This means that a further analysis might be required for future works on other parameters, maybe just pressure value will be more effective in quantifying the recover. The increase instead of y-component for both the cases may be associated to stream tube expansion in order to naturally recover the pressure.

6.2 Surge load cases analysis

In this section a similar analysis to the fixed case one is presented, with focuses both on comparison with experimental data and with fixed case simulation outcomes, which has showed to be coherent with experimental output and wind turbine fundamental theories, allowing it to be considered as the benchmark case.

6.2.1 Loads comparison with experimental data and fixed turbine cases

First of all of at integral forces level, as one can expect from system dynamics, once a system is excited at a certain frequency its response will be at the same frequency. A mechanical system response is characterized by an amplitude and a phase, in this work it was decided to quantify the degree of unsteadiness by measuring the phase (which represents the delay in the response) that thrust signal has with respect to the velocity of the prescribed motion applied to the turbine. Since the oscillation was artificially imposed to the rotor, the harmonic displacement was constructed with phase equal to zero and consequently, its velocity had a 90° shift. A plot and further deepening will be shown in next section, to compare the system response also with pitch cases.

Load Case	LC2.5	Exp2.5	LC2.7	Exp2.7
Mean Thrust [N]	33.16	36.17	33.19	36.19
Thrust Amplitude [N]	1.75	2.17	0.804	1.08
Mean Torque [Nm]	3.32	3.39	3.32	3.39

Table 6.2. Thrust and Torque average values and oscillation amplitude

In this surge dedicated section not all the results will be presented but just for LC2.5 and LC2.7, to underline the differences at two distinct frequencies with the maximum imposed displacements (for the values make reference to table 5.3. In terms of average values the both experimental and numerical thrust and torque did not change with respect to their fixed counterparts.

The average value did not differ significantly from fixed case, hence mean thrust and output can be considered constant. Furthermore, as happened for the averages also the amplitudes are a bit underestimated by the AL solver and the possible causes were assumed the same for the fixed case. Anyway integral outputs may hide something interesting to the analysis and thus, singular blade loads distribution instead should shed light on the platform motion effect: a moving turbine in fact is expected to interact with its own generated wake and the force consequently are supposed to change also at average level.

Local Analysis

It was decided to not show figure like 6.1 and 6.2 with the average loads and aerodynamic coefficient because mean values were almost identical. The average calculation unfortunately was not able to highlight appreciable discrepancies with fixed turbine cases. There emerged the need of focus on parameter capable to show the degree of scattering and fluctuation of these quantities along the three blades. It was decided thus to introduce a synthetic quantity, namely the *Variation Coefficient*, which is defined as the percentage ratio between a quantity standard deviation and its average. This parameter proved indeed to be powerful and suitable, in fact it varied a lot from fixed to moving load cases.

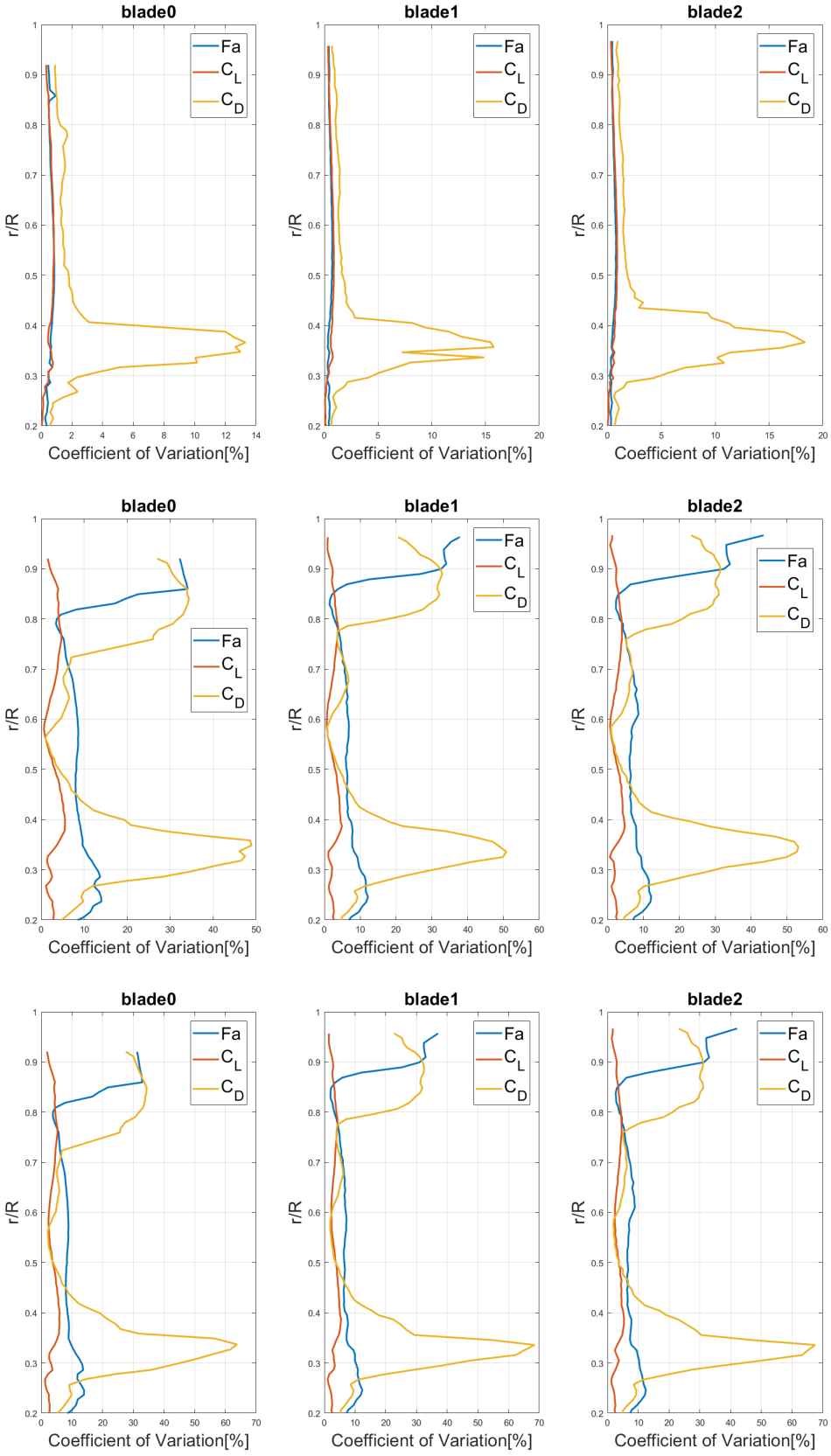


Figure 6.7. Coefficient of variation span distribution for LC1.1 (top) and LC2.5 (middle) and LC2.7 (bottom)

As can be seen from figure **??**, for the steady rated case the Coefficient values are always below 5% for axial force and lift coefficient, while for drag the variation reaches peaks around 15%. Drag coefficient has very low order of magnitude values (10^{-2}-10^{-3} and in the hub region of the span the profile are more prone to stall, leading thus to sudden increase of the coefficient. However this oscillation should not worry, the contribution at low span position is less relevant than in primary and tip regions. On their side the LC2.5 and LC2.7 coefficients of variation get to the value of 30/40% for the axial force in the tip region, this might be associated to turbine interaction with his wake: in the wake the velocity is significantly reduced and if the profile enters the region it may generated low lift and consequently thrust. Between the two frequencies instead the trends are very similar at least for loads.

Dynamic induction extraction

The analysis of dynamic induction proved to be consistent in Mancini's Thesis ([**?**] with the AAT approach. A premise: when dealing with a fixed turbine case the 0.1D upstream and downstream distance of the probing planes, of course, do not change in time. When instead this technique is imported in moving rotor case, like this one, the distance of sampling varies with time and this might be problematic if the surge amplitude becomes larger. Hence this would suggest the insertion at different distances of the probing planes and sample the velocity in a pre-selected section according to the rotor position. For this case study the maximum surge amplitude was equal to 0.035mm (LC2.5), this means that the ratio of A_S and probing line distance (0.1D=0.238mm) was about 15%, and the minimum distance and maximum distance was supposed to slightly impact the measures. In fact in preliminary analysis different probes planes were tested, more in detail 0.05D, 0.1D and 0.15D: the first plane output seemed to be strongly affected by the blade passage, while the third one to loose the requirement of probing on the same flux line (upstream velocity was almost undisturbed and the downstream one completely advected in the wake). Hence a measuring plane whose distance from the rotor which varies from a minimum of 0.085D and a maximum of 0.115D was considered acceptable.

According to what is reported in figure 6.8, surge motion induces an oscillation in the thrust, which is translated in a time dependent axial induction. The maximum value sampled for the induction is half of the threshold indicating the breakdown of linear momentum theory, hence the link between induction and thrust expressed in equation 1.3 holds. Induction time variation highlights the response in terms of amplitude and frequency to the rotor prescribed motion: A_S is larger in LC2.5 and consequently also induction oscillation magnitude is, on contrary LC2.7 f_S is twice the LC2.5 one and the oscillation period is halved. In time history a constant disturbance is visible, this is the 4Hz effect of blade passage due to turbine rotation.
Another factor evidencing that this motions (with these values of amplitude and frequency) effects can be described with linear momentum based models is the mean induction distribution along the blade span.

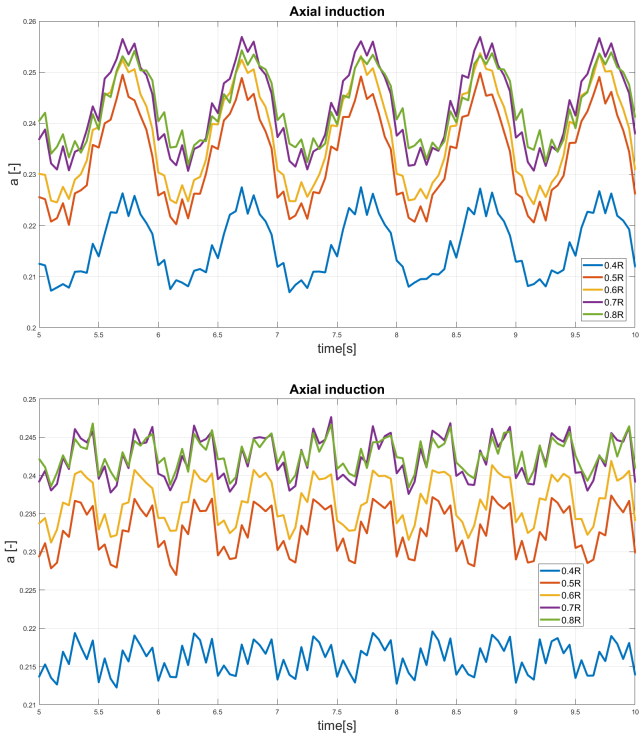


Figure 6.8. Dynamic induction evaluation for LC2.5 (top) and LC2.7 (bottom)

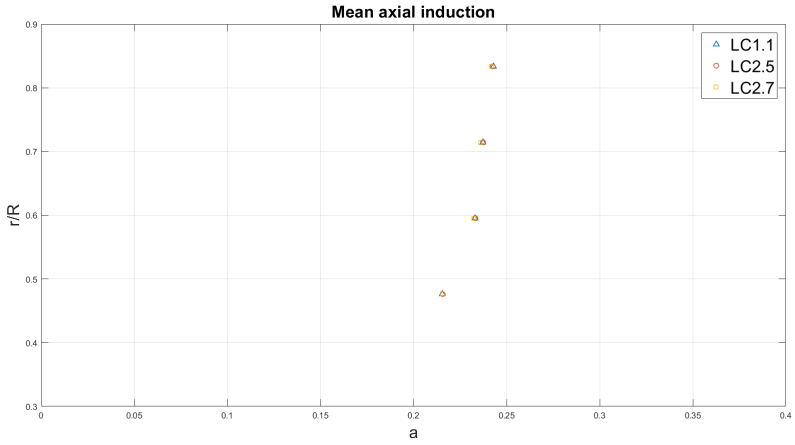


Figure 6.9. Mean axial induction distribution along the blade comparison

Another factor evidencing that this motions (with these values of amplitude and frequency) effects can be described with linear momentum based models is the mean induction distribution along the blade span (6.9). Mean values are almost identical, like the Thrust average outputs.

6.2.2 Wake analysis

Available tests and simulation transverse wake measurements are compared in this section, and then an "internal" comparison of wake evolution is performed for the numerical simulations.

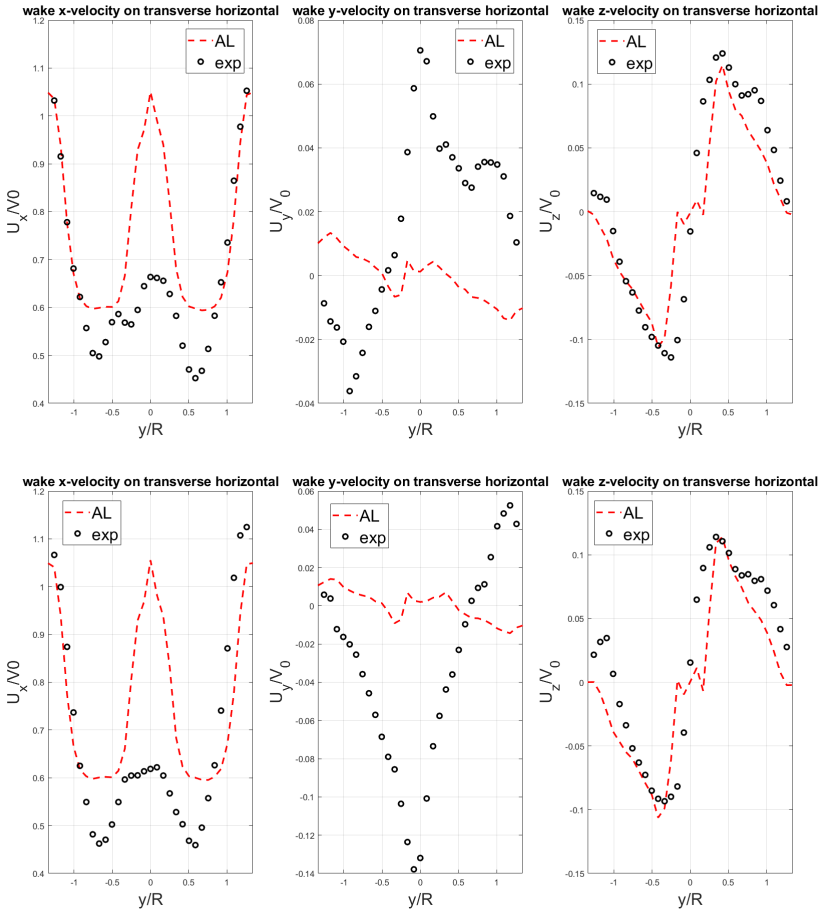


Figure 6.10. Wake numerical and experimental comparison for LC2.5 (top) and LC2.7 (bottom)

As can be noticed in figure 6.11 the wake profile is very well reconstructed for L2.5: neglecting the tower/nacelle absence the deficit trend almost identical on the x-direction, and also for the vertical one the analogies are clear.Nevertheless, in LC2.7 case the shape is not as accurate as before; moreover the experimental y-direction profile showed very different distribution with respect of all the other cases, this discrepancy may be associated to the highest oscillation frequency of the system that is causing, at least for what is visible from test, to eject some flow outside from the stream tube in the radial direction.
Last, wake evolution along main stream direction is reported. Here there is no need of new comments, the recovery seems to be regular for the both cases with higher homogeneity of velocity profile 5D downstream the rotor.

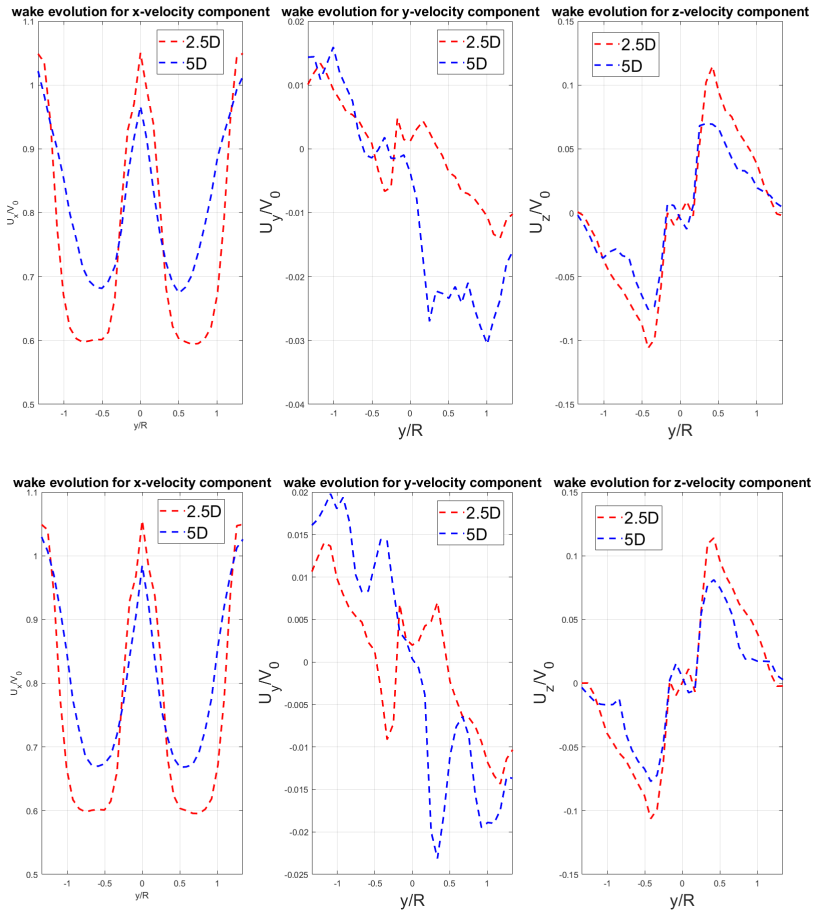


Figure 6.11. Wake axial evolution for LC2.5 (top) and LC2.7 (bottom)

6.3 Pitch and Surge comparison analysis

The first thing that needs to be written is that the criterion applied to select an equivalent pitch motion forcing for the previously analysed surge cases seemed to be consistent. To demonstrate this a phase shift of the thrust response with respect to the motion is reported. A phase (Φ) of the thrust different from zero means that the system needs time to adapt to an input. A phase shift instead equal to zero implies a quasi-steady system response, i.e. immediate. The presence of this lag, is an index of unsteadiness and this is the reason why it is shown here. A note: quasi-static behaviour should have a phase shift of -90° from the hub displacement, since force exchange is associated to momentum (and hence velocity) variation. A value higher than 90° means lag, while lower means lead.

The phases were calculated by means of *fft* (Fast Fourier Transform) function on the thrust signal both for numerical and experimental data. In the figure 6.12 a proportional lag increase with the frequency can be appreciated for GVPM outputs. For what concern instead numerical phase two things have to be deepened: in three out of four cases the delay is very similar for the homologue pitch and surge cases (they are 2.5,2.6 and 2.8). The 2.7 gave a significantly distinct angle. This strong

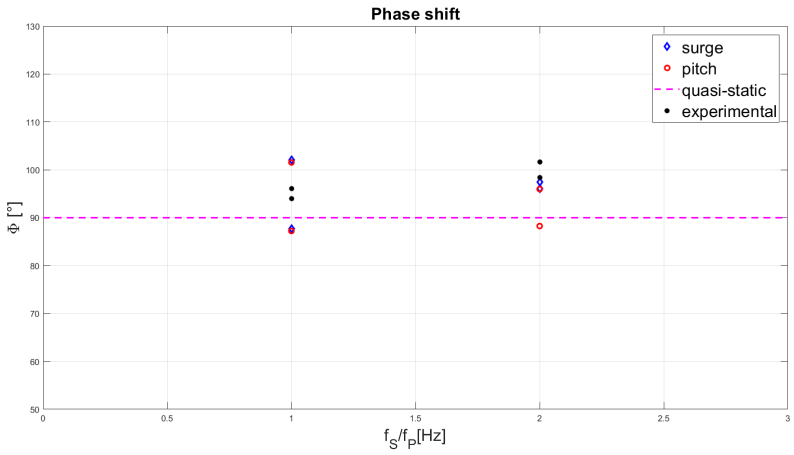


Figure 6.12. Phase shift distribution for the analyzed pitch and surge cases

analogy proved with a certain level of confidence that the reduced frequency criterion is effective. The main drawback of this approach is referred to one fact: $_P$ is the product of three parameters (A_P,f_P,H_P). Keeping fixed the frequency the other two quantities might be combined in several ways to obtain the same value and this paves the way to two opposite scenarios: a longer distance from pitch axis will imply lower hub inclination and hence a less skewed inflow (this can be identified as the surge-like environment), on contrary a lower height translates into a larger maximum inclination, with consequent sensibly difference in the rotor loading above and below its hub (full-pitch scenario). The second peculiar aspect of these phase outputs is the phase lead of three studied cases. It has to be noticed that the lead magnitude is lower than lag one, hence it may be associated to numerical issues associated to the fft post-processing tool. Another option might be that the system vortexes within the could even induce an anticipation of the response, but this should be accurately demonstrated (the core of this Thesis Work differed from a deep and detailed aero-elastic analysis on the mechanical system dynamic response indeed).

6.3.1 Loads comparison

Also in this section the results will be reported just for two load cases, the equivalent ones of LC2.5 and LC2.7, namely p2.5 and p2.7. The experimental pitch motion test data are not available yet,thus the comparison were performed only among at numerical level.
At integral scale the thrust and torque did not differ from the fixed cases and surging cases, even if the amplitude of thrust oscillation resulted higher for the pitch cases. This larger amplitude can be explained in this way: the rotor part above the hub overshoot displacement causes a momentum extraction from the flow that weights more than the below part lower exposition (and consequently is lower extraction).

Nevertheless, for pitch motion the local load analysis gained more relevance: the skewed inflow was expected a priori to generate an higher 3D flow (especially at the blade tip) and unbalances on the rotor according to the relative position with respect to the hub. This fact is partially confirmed by the coefficient of variation analysis.

Load Case	LC2.5	p2.5	LC2.7	p2.7
Mean Thrust [N]	33.16	33.19	33.19	33.15
Thrust Amplitude [N]	1.75	2.012	0.804	1.699
Mean Torque [Nm]	3.32	3.33	3.32	3.31

Table 6.3. Thrust and Torque average values and oscillation amplitude for equivalent cases

Local loads analysis

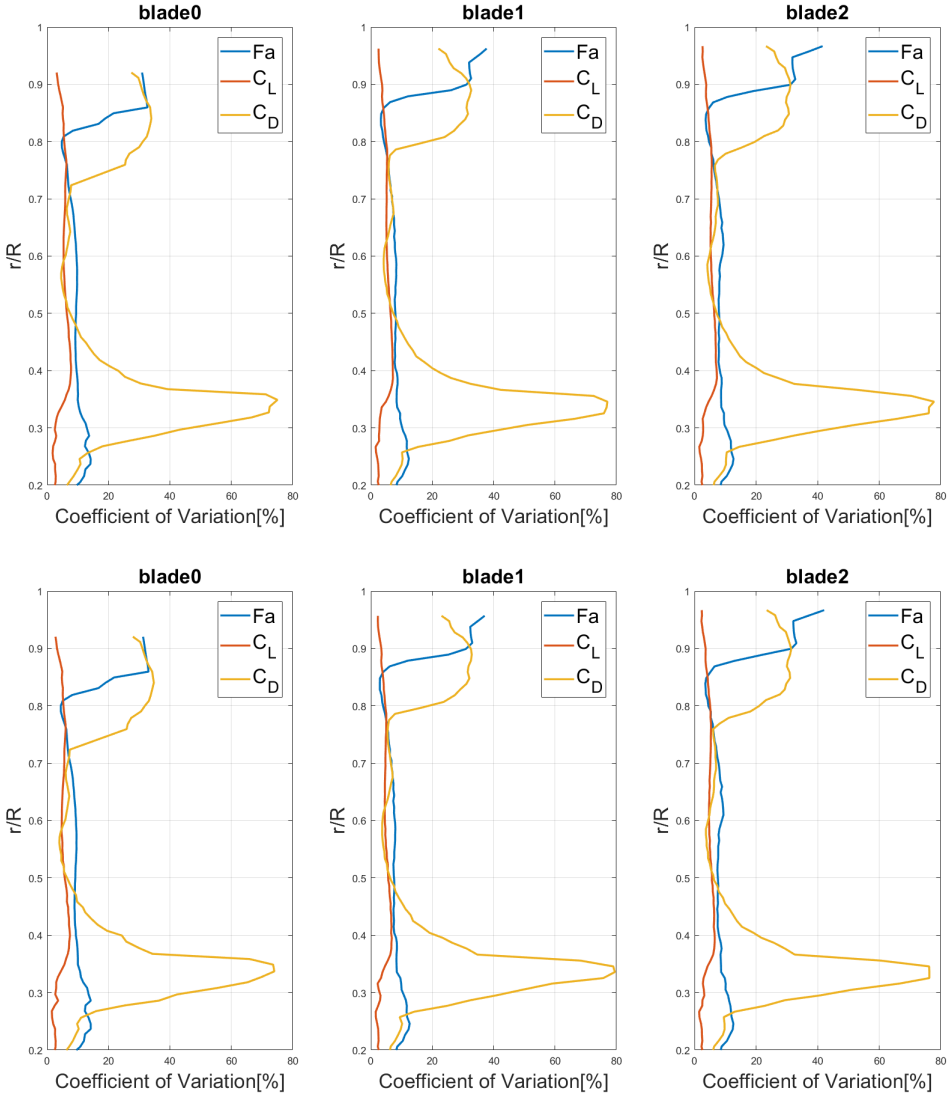


Figure 6.13. Coefficient of variation distribution for pC2.5 (top) and pC2.7 (bottom)

The figure 6.13 underlines that the dispersion of this quantities at the tip were reduced compared to their respective surge counterparts (figure 6.7). In few words: during a pitching oscillation period the blade tip, due to the rotation, will pass from above to below relative position with respect to the hub; and thanks to this combined motion

it will be less probable to be at the maximum amplitude displacement. While for the surge cases the tip is always subjected to the biggest (and homogeneous) motion amplitude which makes bigger the dispersion of tip vortexes generation region. To counterbalance this lower dispersion it has to be noted instead that the C_D variation seems to be higher (up to 20% more) and this caused an increase of the inefficiency of the machine.
In terms of loads this kind of pitch motion resembled more a surge-like scenario than a full-pitch one and this may be good, because the dynamic inflow models applied for the linear displacements could be exploited also for a pitching case. The reliability of this scenario was supported also by the null value of radial force on the blades.

Dynamic induction extraction

The premise of AAT technique is the work with a purely axial inflow, and a pitch motion does not satisfy this requirement. However for the studied load cases, especially for the p2.7, the inclination was so small to be neglected (with a 0.6° pitch amplitude in fact the maximum variation of vertical coordinate is equal to the 1.04%). This evaluation was performed in order to compare the dynamic induction output of two equivalent pitch and surge cases.

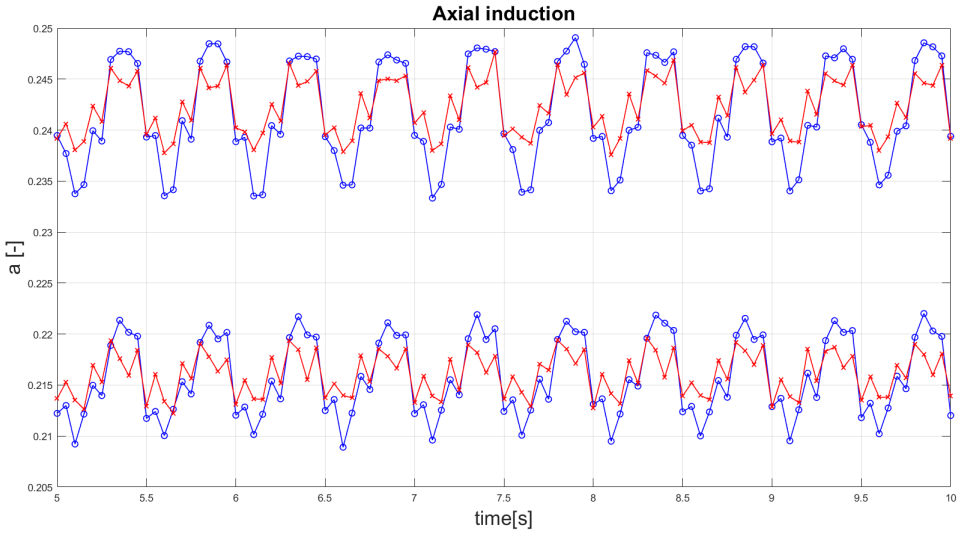


Figure 6.14. AAT outputs for LC2.7 (red) and p2.7 (blue) load cases

In figure 6.14 the induction time history is reported just for two distinct blade positions (0.4R below and 0.7R above) to avoid superposition of the lines. Also in this case the linear momentum conditions are satisfied and hence the induction describes the thrust history in a reliable way, in fact larger induction amplitude for the pitch cases is consistent with larger thrust oscillations.

6.3.2 Wake Analysis

For floating wind turbines, one of the most profitable solution the exploit scale economy is to built wind farms. In these clusters of turbines one of the most relevant requirements is the good characterization of the wake downstream a rotor, in order optimize the turbine positioning.
In this section the reported output are associated only to x-direction velocity profile, since it is the most significant component associated to the thrust on the machine. Moreover, y-direction was always one order of magnitude lower than the other ones while z-component proved to be reliable for the majority of studied cases.
Also for wake analysis was expected a more skewed velocity profile in pitch motion compared to surge one, that is the reason why also a vertical transverse measurement set-up was included.

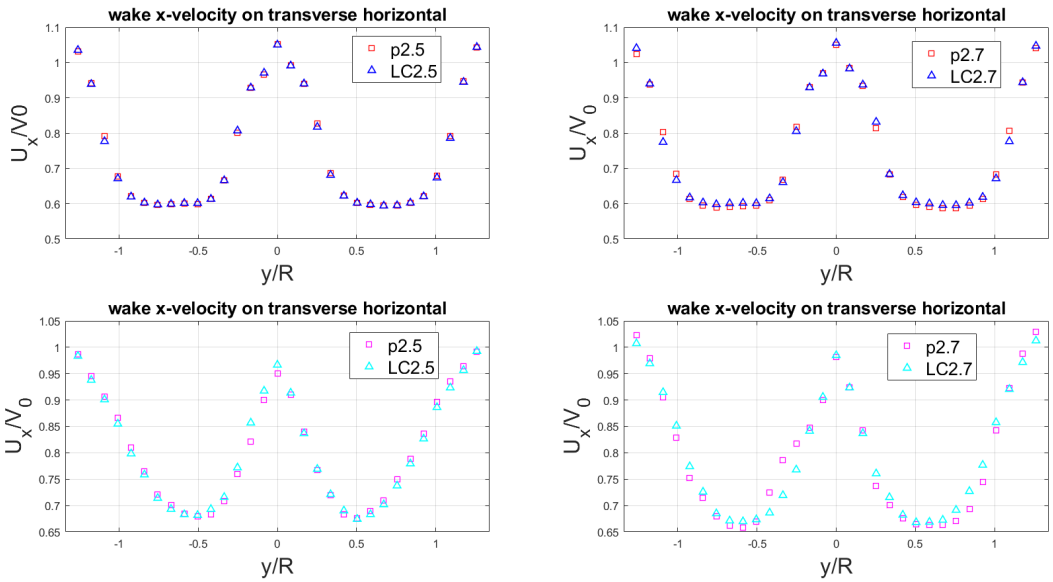


Figure 6.15. Wake evolution comparison for equivalent pitch/surge cases measured with an horizontal transverse, both 2.5D (top) and 5D (bottom) downstream the rotor

At the horizontal transverse the differences between pitch and surge compared cases almost negligible in x-direction. The outputs are averaged but it is impressive to have this profile similarity, the only appreciable distinction is 5D downstream the turbine for the p2.7 case in which an asymmetry is present (bottom right frame in figure 6.15), maybe due to vortex instability in that region when excited at a higher frequency. For all the cases there is a good recovery and homogenization of the profiles. At vertical transverse instead a wake asymmetry is present, but not only for pitch load cases. Also surge features this characteristic even if in lower grade. It his interesting to notice that the profile mismatch is more significant 5D downstream the rotor and this is a relevant aspect for floating wind farms: the pitching motion influence is trailed by the vortexes in the wake and this effect is relevant also far downstream the turbine, causing high load unbalances on the following turbines and a consequent deterioration of the performances and fatigue life resistance

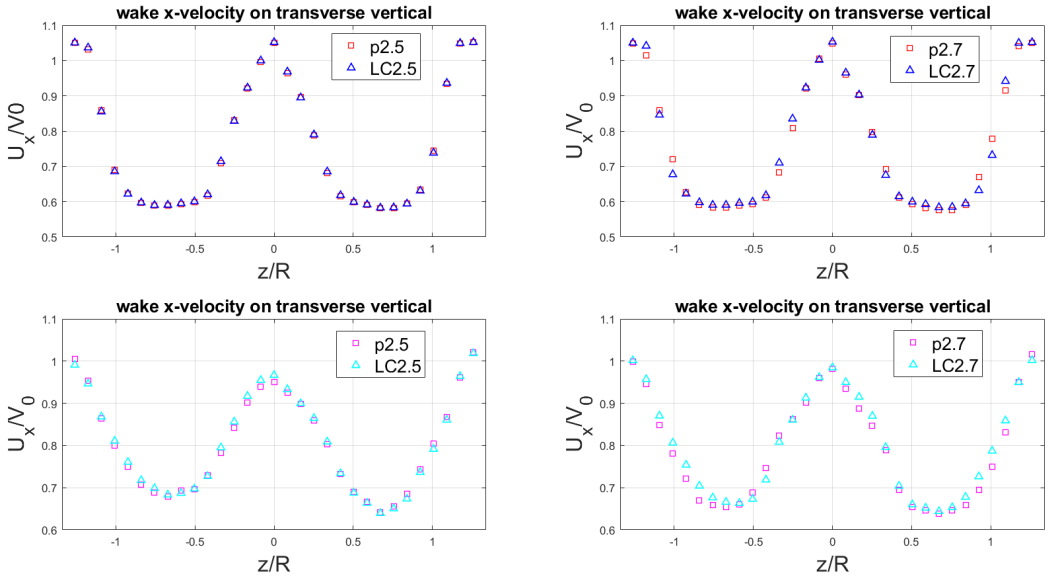


Figure 6.16. Wake evolution comparison for equivalent pitch/surge cases measured with a vertical transverse, both 2.5D (top) and 5D (bottom) downstream the rotor

6.4 AL pitch simulation results comparison with other codes

Thanks to the collaboration with Mancini, a comparison among three different codes was made available. The three codes were the in-house developed AL code (indicated as 'AL'), a BEM codes from TNO Institute ('BEM') and a Free Wake Vortex Method always from TNO ('TNO'). Due to the lack of pitch motion experimental outputs it was performed a comparative analysis among the three codes to see how they described the platform motion effects.

6.4.1 Integral Thrust and Torque

First of all it is interesting to see how the three numerical tools estimated the mean thrust and torque, the amplitude of thrust oscillation and the phase shift of the thrust response with respect to the pitch frequency displacement (results are reported just for p2.5 and p2.7).

p2.5	AL	TNO	BEM
Mean Thrust [N]	33.19	33.78	32.99
Thrust Amplitude [N]	2.012	2.509	2.496
Mean Torque [Nm]	3.33	2.69	2.52

Table 6.4. Thrust and Torque average values and oscillation amplitude for different codes in p2.5

p2.7	AL	TNO	BEM
Mean Thrust [N]	33.15	33.77	33.00
Thrust Amplitude [N]	1.699	2.216	2.272
Mean Torque [Nm]	3.31	2.69	2.52

Table 6.5. Thrust and Torque average values and oscillation amplitude for different codes in p2.7

As can be seen the average values of Torque and Thrust do not change significantly from, it is more interesting to notice that that the BEM and TNO codes showed an higher amplitude in terms of thrust oscillation. This facts could be addressed to the discrete nature of the CFD computations, in which the line motion might not be fully captured on the grid (for example if the line does not cross completely a cell the intersection of the line will be performed always in the same cell), moreover, the system of vortex generated at the the tip in not regulated by the presence of correction and is not artificially created from it since it is simulated and not artificially implemented, causing a more random load on the blade.

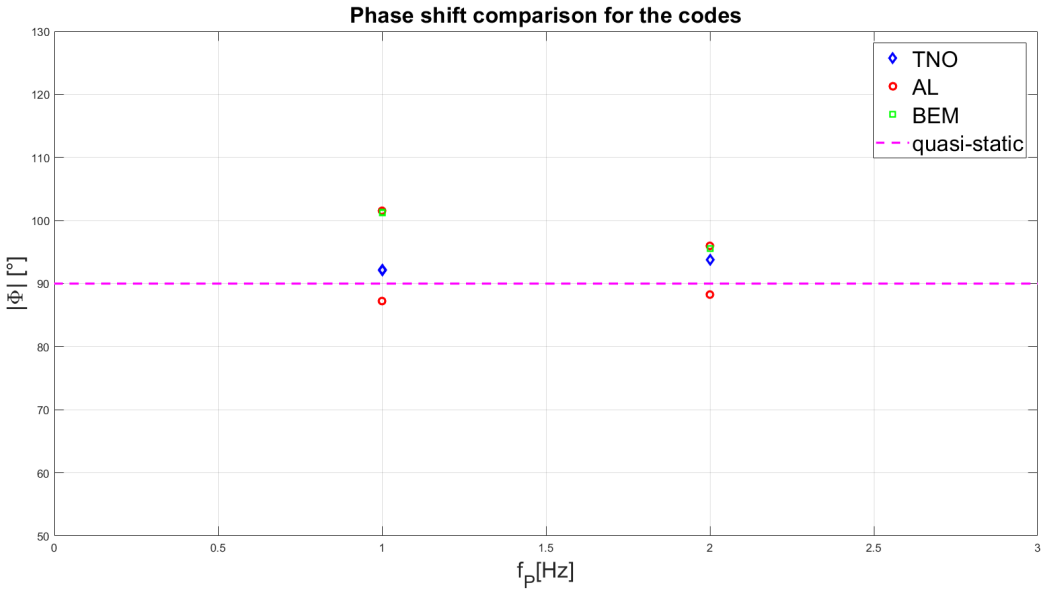


Figure 6.17. Phase shift comparison among the codes

The AL code experienced a phase lead in the two cases with larger amplitude (p2.5 and p2.7), this seemed so strange since the only evidences of this phenomenon are reported in a TNO report ([44]), in which the the phase lead is detected only in codes featuring a dynamic stall model. Further analysis could be more clear results of this phase value which maybe is caused also by the spreading of this effect at multiple frequencies of the motion due to the sampling discretized domain.
Looking instead at more consistent results of the AL, it seems that the phase lag is detected like in the BEM code, this is maybe due to the similarity of the two method in terms of force evaluation combined with a lower oscillation amplitude

which results in a "more steady" inflow condition. For what regards instead the TNO code, it experiences an increasing delay with the frequency (a coherent trend with the experimental surge outputs of figure 6.12, even if the two motions are different their similarity was almost proved by the reduced velocity indicator).

6.4.2 Local forces evaluation

The, at least for now apparent, underestimation of thrust oscillation amplitude might be explained also in term of local loads.

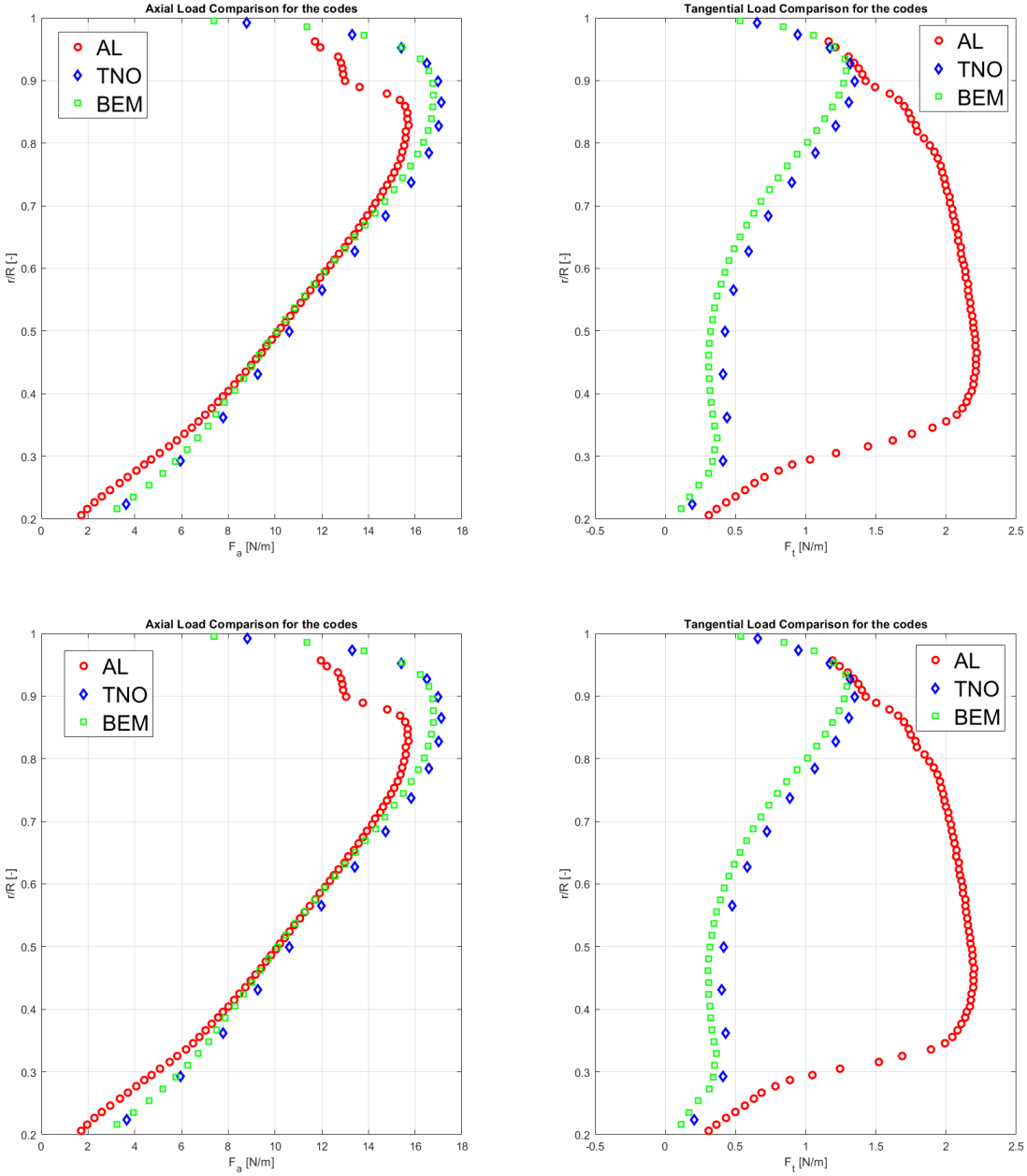


Figure 6.18. Axial (left) and tangential (right) mean load distributions for p2.5 (top) and p2.7 (bottom)

At average level the axial load are identical for most part of the span, there a significant difference just at the tip, referable to the CFD vortex presence. On the contrary, significant mismatch was obtain in terms of mean Torque, and the tangential blade loads highlight this fact. Hence, even if whole codes underestimate the thrust maybe for the obstacle effect of real blades, nacelle and tower which increases the module of inlet velocity, the average torque level is accurately reconstructed by AL only (to have full confidence, a check with tangential loads distribution might be done with experimental data).
Last, to justify the lower amplitude of thrust oscillation the Coefficient of Variation distribution for the blade axial load comes to help.

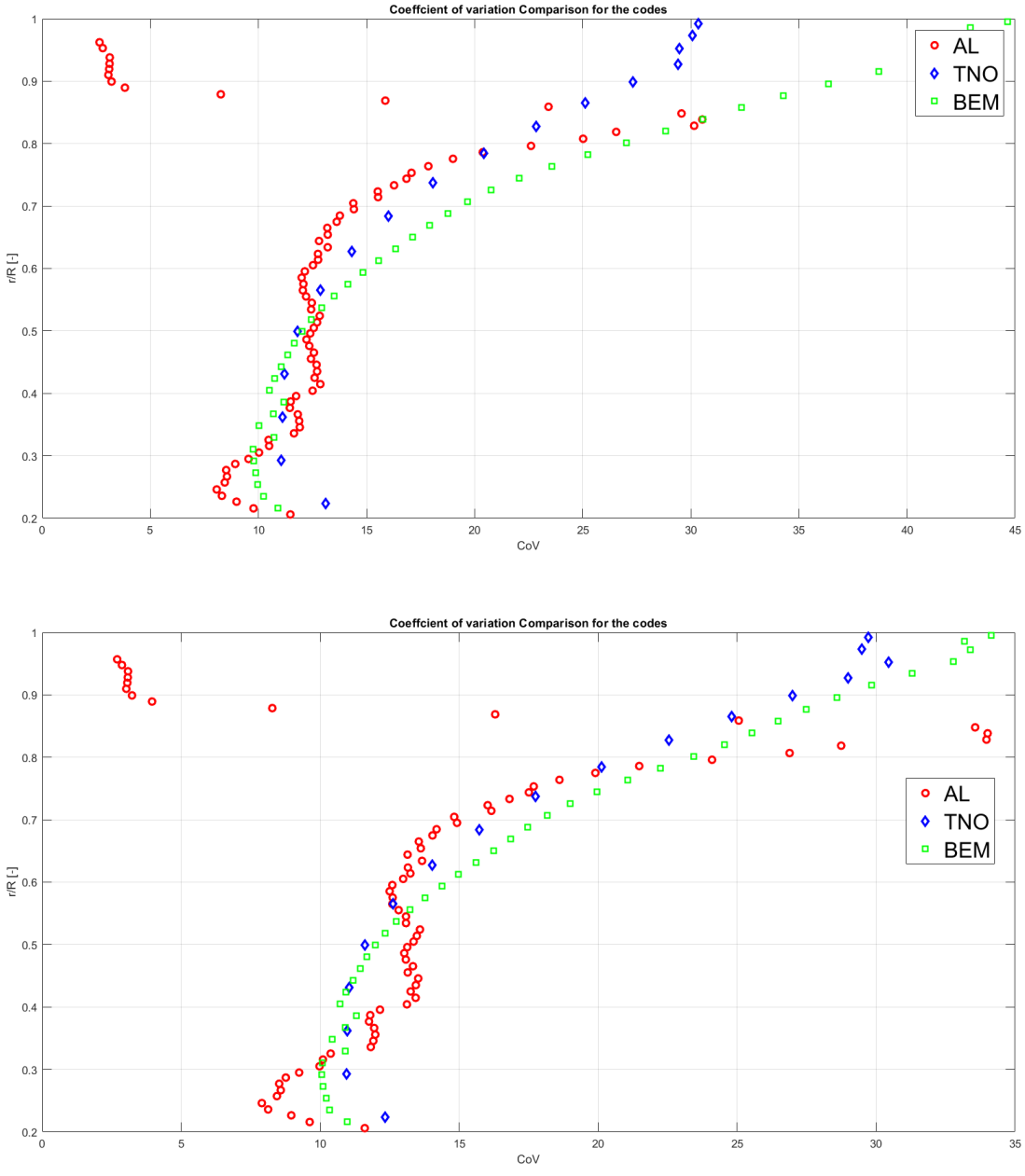


Figure 6.19. Axial force Coefficient of variation distribution for p2.5 (top) and p2.7 (bottom)

As one can notice from figure 6.19, the distribution of the variation trends are similar for the three codes (except for less regularity of the AL one) apart from the tip value. For AL code the value experiences a significant drop: even if the oscillation amplitude is the highest at tip, it seems that at that span position the force dropped so much to perceive sensible oscillations. Since there's no an experimental counterpart it is difficult to state with certainty which trend is the most realistic. The drop is likely to occur also for BEM and TNO but just at the very end. With an high displacement the Reynolds regime may change significantly and the AL is not able to consider this variation, moreover the rotating actuator line intersection with the mesh grid is not the same every time step, especially at the tip the number variation is higher, causing then a zero force at the apex.

Conclusions and Future Works

6.5 Conclusions

In-house developed AL code proved his reliability in describing platform motion effects on wind turbines unsteady aerodynamics. Of course this Thesis Work focused on restricted range of frequencies and amplitudes and did not stress too much the code simulating prescribed motions that would lead to the Linear Momentum theory validity limits. Even if interesting, the simulation of extreme loads may be useful to fix the threshold above which the code will not work anymore but at the same time it will be describing hardly realistic operating conditions.
One of the aims of this work was the implementation of platform pitch motion of the turbine to allow a simulation database ready to be compared with experimental tests which will be performed at POLIMI for the OC6 campaign, and it was fully achieved: the turbine hub was verified to move according to the desired trajectory and velocity. After the implementation the turbine was allowed to move in surge or pitch motion and hence the necessity of the characterizing each displacement effect on the turbine performances arose. Although at a first look the two motions seem to impact in a completely different way on the flow, the compared quantities, at least at average level of approximation, shed a light more on their analogies in terms of loads.
The loads on three blades followed a similar pattern: the ensamble average used to described the forces distribution highlighted that the blade starting from vertical position (*blade0* intersected a lower minimum number of mesh cells, this lead to a reduction of the tip actuator points, which might have impacted in part for the load underestimation.It is suggested to implement an interpolation routine in the code for a fixed number of actuator line points to give the possibility of comparing blade performances and loads with the same number of elements for each of them.
The criterion applied to find equivalent surge and pitch cases demonstrated to be consistent for the majority of the analyses. However reasoning on the very simple ΔV equation there emerged the possibility of entering in possible concurrent scenarios. Looking at the Thesis outputs one may think that in most of the cases a pitch motion will impact on the turbine performances and wake generation like a surge displacement does as much as their reduced velocities are similar.This statement cannot be confirmed until a further check is done: $\Delta V_P = 2\pi f_P A_p H_p$ expression intrinsically contains the "degree of pitch" of a pitch motion.Assuming that a certain reduced velocity is assigned, once the distance from the pitch axis increases the amplitude of pitch oscillation will reduce. The ideal limit case is the one in which pitch axis center of rotation distance tends to infinity, this would means a rotation around infinitely distance center of rotation, namely a linear displacement identified as

surge. In reality this will never happen but it might be interesting to set a threshold value to distinguish between a purely pitch effect or a surge-like condition. This is not just a proof of concept, in fact in floating wind turbine the pitching axis is not always the same and thus there might be sea wave conditions for which the effects can be attributed to surge, this would imply the adoption surge dynamic inflow models for the designed controllers.
As already stated the mean performances, especially in terms of loads, did not suffer too much the platform motion effects. Their dispersion was instead quantified by means of coefficient of variation around the mean out, showing scattering where the vortexes presence was relevant (namely at the blades hub and tip). Anyway, given these quite impressive analogies in terms of loads the pitch load cases wake proved to suffer of asymmetric distribution, especially, far downstream the turbine rotor for higher frequencies platform motion. This effect in single turbine system would not be interesting, while in a floating wind farm environment should impact significantly on performances.

6.6 Future Works

The Thesis work focused mainly on the study of the code an its implementation. In future works might be interesting to deepen in the set-up of the numerical schemes and solution of the CF problem. First of all the schemes were taken for granted basing on the good results of previous works but a study on different order divergence schemes might be useful to see how the vortex system is captured. A more sophisticated work could also be done on the solver, which is now explicit: one of the most important phenomenon associated to unsteady aerodynamics is the dynamic stall, in which the polars becomes for brevity time dependant. The nature of this phenomenon clashes with an explicit formulation and hence an implicit solver may be adopted in future. The boundary conditions imposed were very simple, but for a reasonable cause, namely avoiding the instability of LES methods at the boundaries. At inlet a clean velocity profile was modeled, however In the GVPM there are measurements for getting a more accurate inlet flow profile which may be imported then in the boundary conditions; alternatively a precursor simulation could be run.
For what concerns the code itself, different implementations can be made, starting from a simple addition Reynolds interpolation function or with a more complex dynamic stall implementation. It could be useful to add one of the cited tip corrections in general to see how much performance evaluation improves with respect to the normal code. Another suggested article for this scope could be the application of a filter to the inflow ad proposed by Martìnez-Tossas, [45]. It is true that an HAWT blade section operates almost at fixed Reynolds number, but the platform motion, if high, may change heavily the turbulence conditions, especially at low Reynolds numbers, in which polars are more sensible to this non dimensional quantity. Last code implementation may be the Herràez sampling technique([42]) for the extraction of the rotor induction in order to compare different approaches.
Last, to see AL performances with a wind farm framework it would be interesting to simulate a turbines with the inlet conditions coincident with the outlet a turbine, in order to tune and calibrate optimal positioning.

Appendix A

First Appendix

A.1 List and Description of AL dictionaries

In turbine/actuatorForceLineBaseClass/**actuatorForceBaseClass**

- **actuatorForce.h**:
  Description
  actuatorForce solver. It inserts a volumeForce in a cell of the domain. The magnitude and the angle of the volume force are calculated following the EVM approach, i.e. taking the relative velocity the one calculated along a reference line positioned perpendicular to relative velocity direction, some chords wide and some chords upstream respect to the effective force application point. Based on this point, the start and end points of reference line are now calculated in run-time. The sampling velocity on the reference line is calculated by CalcRefVelocity with intersectionData class. Then, a correction based on the interpolation of U for the effective force point is performed, if needed.

  include "intersectionData.h"
  include "interpolation.H"
  include "fvMeshSubset.H"

- **actuatorForceApplyVolumeForce.cpp**:
  Description
  Adds a volume force in a cell of domain

  include "actuatorForce.h"
  include "intersectionData.h"

- **actuatorForceConstructor.cpp**:
  Description
  Class constructor – destructor.

  include "actuatorForce.h"

- **actuatorForceGetFunction.cpp**:
  Description
  Getter functions.

  include "actuatorForce.h"

- **actuatorForceReadForce.cpp:**
  Description
  Read problem data dictionaries.

  include "actuatorForce.h"

- **actuatorForceReferenceVelocity.cpp:**
  Description
  Calculates from the velocity field the reference velocity needed for the interpolation of airfoil polar tables and hence the aerodynamic forces. In this case the reference velocity is the mean value of velocity along a line defined the start and the end points.

  include "actuatorForce.h"
  include "intersectionData.h"

- **actuatorForceRegKer.cpp:**
  Description
  Distribute force trough a spatial function

  include "actuatorForce.h"

- **actuatorForceVelInterp.cpp:**
  Description
  Interpolate velocity field data in the desidered point

  include "actuatorForce.h"

- **intersectionData.cpp:**
  Description
  Tool for find intersection between mesh and line.It returns the length of the line, a list of the intersected points, a list of all points of interest (intersected points and start and end point of the line), a list for midpoints between intersected points. The constructor of this class requests as inputs the meshcase and the boundBox referred to that mesh.

  include "intersectionData.h"

- **intersectionData.h:**
  Description

See **intersectionData.cpp**

include "fvCFD.H"
include "treeDataFace.H"
include "indexedOctree.H"
include "Random.H"
include "Pstream.H"

in turbine/**actuatorForceLineBaseClass**

- **actuatorLine.h:**
  Description
  actuatorLine class. It inherits actuatorForce class methods. It finds the actuatorLine points (that represent the blade) and for each of them it lets the calculation of corresponding reference velocity as stated by EVM.

  include "actuatorForce.h"
  include "intersectionData.h"
  include "actuatorLineInterpolate.h"

- **actuatorLineInterpolate.h:**
  Description
  Interpolation of lists (origin: SOWFA code).

- **actuatorLineApplyLineForce.cpp:**
  Description
  It calculates aerodynamic force along one blade. It uses actuatorForce::calcVolForce for apply volumeForce on founded intersections.

  include "actuatorLine.h"

- **actuatorLineCalcLineForce.cpp:**
  Description
  Virtual function.

  include "actuatorLine.h"

- **actuatorLineConstructor.cpp:** *Description*:
  Class constructor – destructor.

  include "actuatorLine.h"

- **actuatorLineEvalRefVel.cpp:**
  Description
  Virtual function.

  include "actuatorLine.h"

- **actuatorLinefindMeshIntersection.cpp:**
  Description
  It finds the intersection between the mesh and the actuatorLine representing the blade.

  include "actuatorLine.h"

- **actuatorLineGetFunction.cpp:**
  Description
  Getter functions.

  include "actuatorLine.h"

- **actuatorLineUpdateLinePos.cpp:**
  Description
  Virtual function.

  include "actuatorLine.h"

In turbine/HAWTalfaDYNpitchGENERIC/**HAWTurbine**

- **HAWTurbine.h**
  Description
  Class representing the horizontal axis wind turbine

  include "fvCFD.H"
  include "OFstream.H"
  include "actuatorLine.h"
  include "HAWTblade.h"

- **HAWTConstructor.cpp**
  Description
  Class constructor and destructor.
  include "HAWTurbine.h"
  include "interpolation.H"

- **HAWTForce.cpp:** *Description*:
  Overall turbine thrust, torque and power calculation.

  include "HAWTurbine.h"

- **HAWTReadTurbine.cpp:**
  Description
  Read problem data from turbineProperites and airfoilProperties dictionaries.

  include "HAWTurbine.h"

- **HAWTRotateTurbine.cpp:** *Description*
  Up-date turbine blades position.

  include "HAWTurbine.h"

- **HAWTsubMesh.cpp**:
  Description
  Up-date turbine blades position.

  include "HAWTurbine.h"

- **HAWTWrite.cpp:**
  Description
  Write function.

  include "HAWTurbine.h"

in turbine/HAWTalfaDYNpitchGENERIC/**HAWTbladeLine**

- **HAWTblade.h:**
  Description
  Class representing turbine blade via actuatorLine model. It derives from actuatorLine class.

  include "actuatorLine.h"

- **airfoilTypeIDInterpolation.h**
- **ClCdInterpolation.h**
- **cart2bld.h**
- **calcForce.h**
- **HAWTbladeApplyBladeForce.cpp:**
  Description
  Function that summarizes the procedure to force application over the actuatorLine.

  include "HAWTblade.h"

- **HAWTbladeCalcForce.cpp:**
  Description
  Calculation of the force, thrust and torque for the actuatorLine.

  include "HAWTblade.h"

- **HAWTbladeCollectData.cpp:**
  Description
  Write function, for postProcessing utilities.

  include "HAWTblade.h"

- HAWTbladeConstructor.cpp:
  Description
  Class constructor – destructor.

  include "HAWTblade.h"

- **HAWTbladeDataInterp.cpp:**
  include "HAWTblade.h"

- **HAWTbladeEvalRefVel.cpp:**
  Description
  Reference velocity calculation function for each actuatorLine point.

  include "HAWTblade.h"

- **HAWTbladeGetFunction.cpp:**
  Description
  Getter functions.

  include "HAWTblade.h"

- **HAWTbladeLineForce.cpp**
  Description
  It calculates aerodynamic force along one blade. It uses actuatorForce::calcVolForce for apply volumeForce on founded intersections.

  include "HAWTblade.h"

- **HAWTbladeReadBlade.cpp:**
  Description
  Read dictionary function.

  include "HAWTblade.h"

- **HAWTbladeRegKer.cpp:**
  Description
  Distribute force trough a spatial function

  include "HAWTblade.h"

- **HAWTbladeSetParameters.cpp:** *Description*
  Set lists of parameter for the actuatorLine.

  include "HAWTblade.h"

- **HAWTbladeUpdateLinePos.cpp:**
  Description
  Up-date actuatorLine position according to time and rotational speed.

  include "HAWTblade.h"

- **HAWTbladeWrite.cpp:** *Description*
  Write function, for postProcessing utilities.

  include "HAWTblade.h"

A.2 Pitch motion in AL code implementation

in constant/**turbineproperties** :

- PitchFreq 0;
- PitchAmpDeg 0;
- H 0;

in turbine/HAWTalfaDYNpitch/HAWTurbine/**HAWTConstructor.cpp** :

- PitchDir(0,0,1.0)

in turbine/HAWTalfaDYNpitch/HAWTurbine/**HAWTReadTurbine.cpp** :

- PitchFreq=scalar(readScalar(turbineProperties.lookup("PitchFreq")));
- PitchAmpDeg=scalar(readScalar(turbineProperties.lookup("PitchAmpDeg")));
- H=scalar(readScalar(turbineProperties.lookup("H")));
- PitchAmp=Foam::constant::mathematical::pi*PitchAmpDeg/180;

in turbine/HAWTalfaDYNpitch/HAWTurbine/**HAWTRotateTurbine.cpp** :

- PitchAng=PitchAmp*sin(2*Foam::constant::mathematical::pi*PitchFreq*runTime.value());

- ddtPitchAng=2*Foam::constant::mathematical::pi*PitchFreq*PitchAmp*cos(2*Foam::constant::mathematical::pi*PitchFreq*runTime.value());

- hubVel=H*cos(PitchAng)*ddtPitchAng*SurgeDir-H*sin(PitchAng)*ddtPitchAng*PitchDir;

- hubPosition=hubPositionStart+H*sin(PitchAng)*SurgeDir+H*(cos(PitchAng)-1)*PitchDir;

in turbine/HAWTalfaDYNpitch/HAWTurbine/**HAWTurbine.h** :

- //- vector platformPitch Direction.
  vector PitchDir;

- //- platformPitch freq.
  scalar PitchFreq;

- //- platformPitch amp.
  scalar PitchAmp;

- //- platformPitch amp. in deg
  scalar PitchAmpDeg;

- //- Tower Pitch height.
  scalar H;

- //- platformPitch instantaneius amp.
  scalar PitchAng;

- //- platformPitch instantaneous amp. derivative
  scalar ddtPitchAng;

in turbine/HAWTalfaDYNpitch/HAWTbladeLine/**HAWTblade.h**:

- //- vector platformPitch Direction.
  vector PitchDir;

Appendix B

Second Appendix

B.1 OC6 Load Cases Table for Surge Motion

Load Case	V_0 (m/s)	F_{surge} (Hz)	A_{surge} (m)	Ω (rpm)	θ_P (°)	CW	AW	PIV	Test #
2.1	4.0	0.125	0.125	240	0	✓	✓	✓	33
2.2	4.0	0.125	0.03	240	0	✓		✓	36
2.3	4.0	0.5	0.065	240	0	✓	✓	✓	41
2.4	4.0	0.5	0.015	240	0	✓		✓	44
2.5	4.0	1.0	0.035	240	0	✓	✓	✓	50
2.6	4.0	1.0	0.01	240	0	✓		✓	52
2.7	4.0	2.0	0.008	240	0	✓	✓	✓	59
2.8	4.0	2.0	0.004	240	0	✓		✓	60
2.9	6.0	0.125	0.125	265	12.5	✓		✓	65
2.10	6.0	1.0	0.035	265	12.5			✓	83
2.11	6.0	2.0	0.0125	265	12.5	✓		✓	91

Figure B.1. Defined table for surge oscillation load cases

In the figure B.1, there is the the definition of full test matrix for the OC6 program. The thicks below CW, AW and PIV mean that experimental measurements are required for the specified load case. In this thesis work the analysed cases, at numerical level, were LC2.5, 2.6, 2.7 and 2.8.

Appendix C

Second Appendix

C.1 Surge and Pitch simulations output

C.1.1 LC2.6

Loads

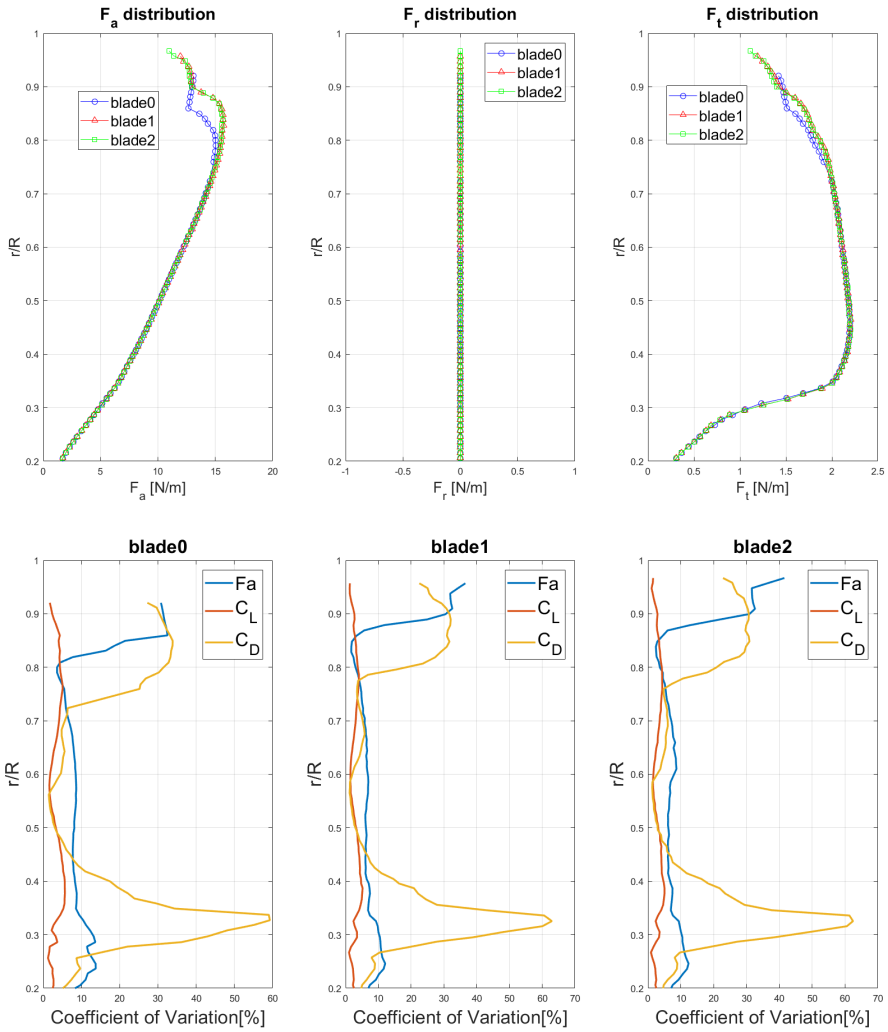


Figure C.1. Average Loads on the three turbine blades (top) and Coefficient of Variation distribution (bottom) for LC2.6

Wake Evolution

No experimental date were available for the LC2.6 wake measures.

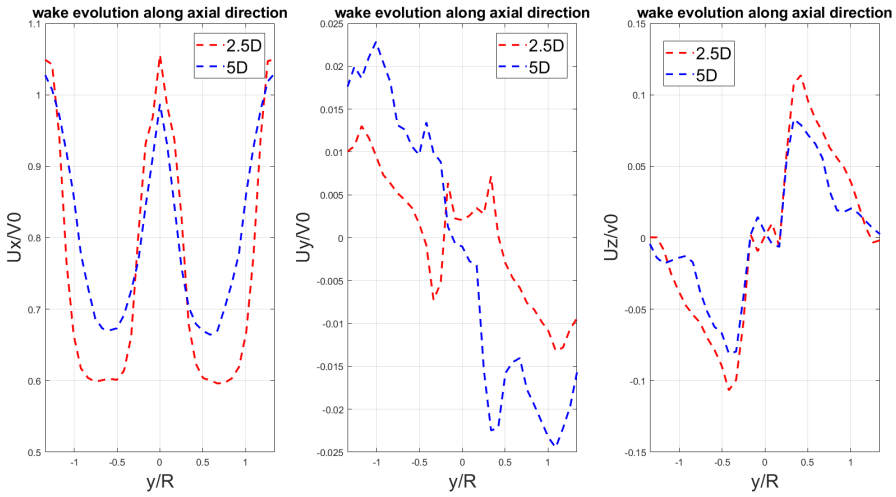


Figure C.2. Velocity profile components evolution, measured by horizontal transverse, for LC2.6

C.1.2 LC2.8

Loads

In this section it was decided to not report mean loads due to their low differences with previous cases.

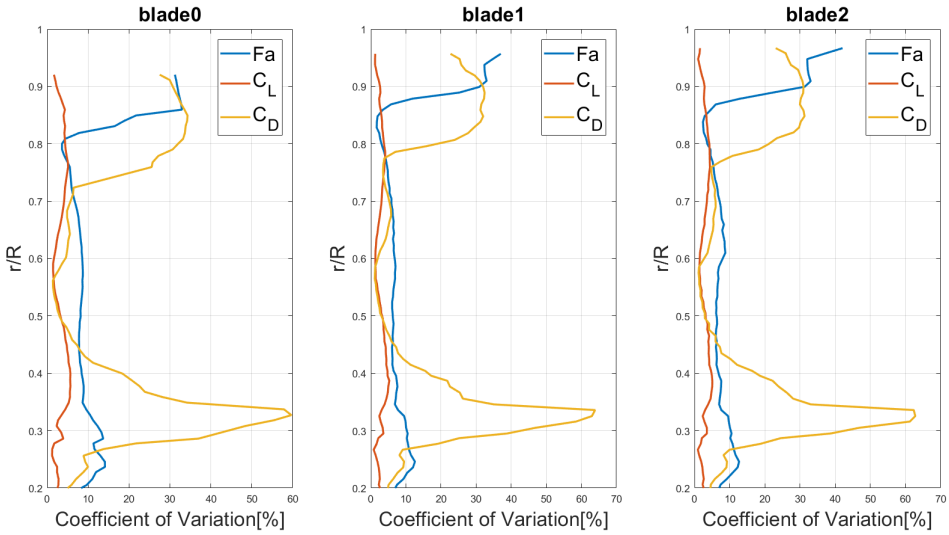


Figure C.3. Coefficient of Variation of loads distribution for LC2.8

Wake Evolution

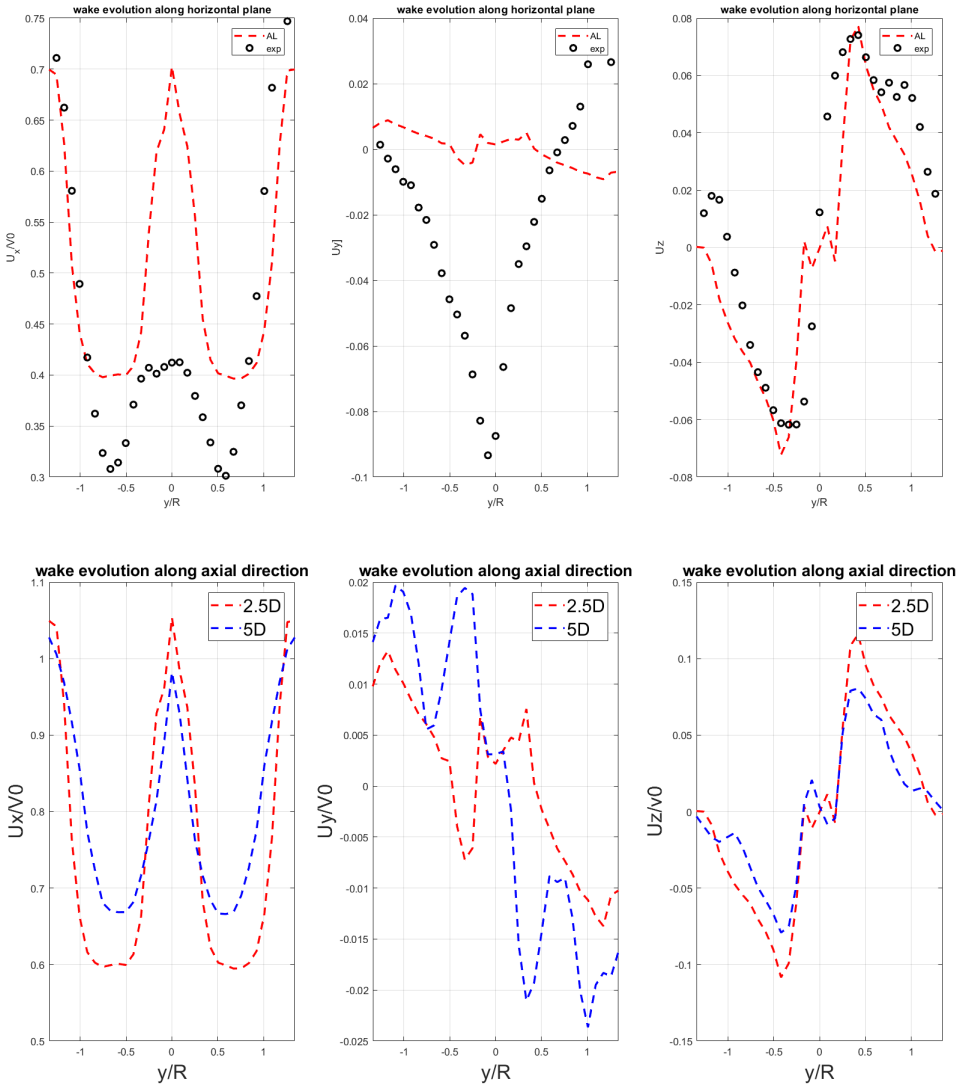


Figure C.4. Velocity profile components comparison with experimental data (top) and numerically estimated wake evolution, measured by horizontal transverse, for LC2.8

C.1.3 p2.6

Loads

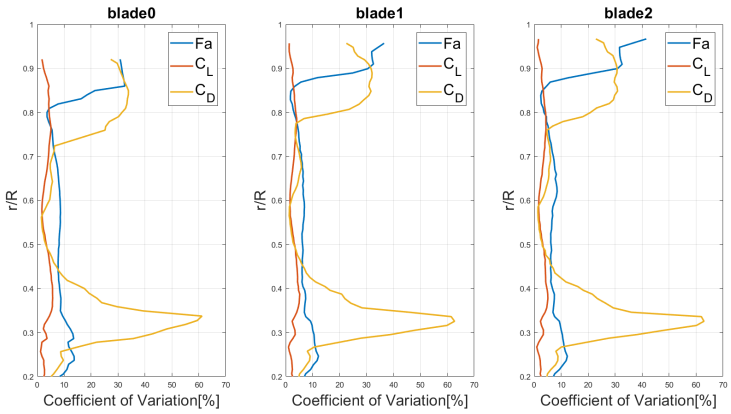


Figure C.5. Coefficient of Variation of loads distribution for p2.6

Wake Evolution

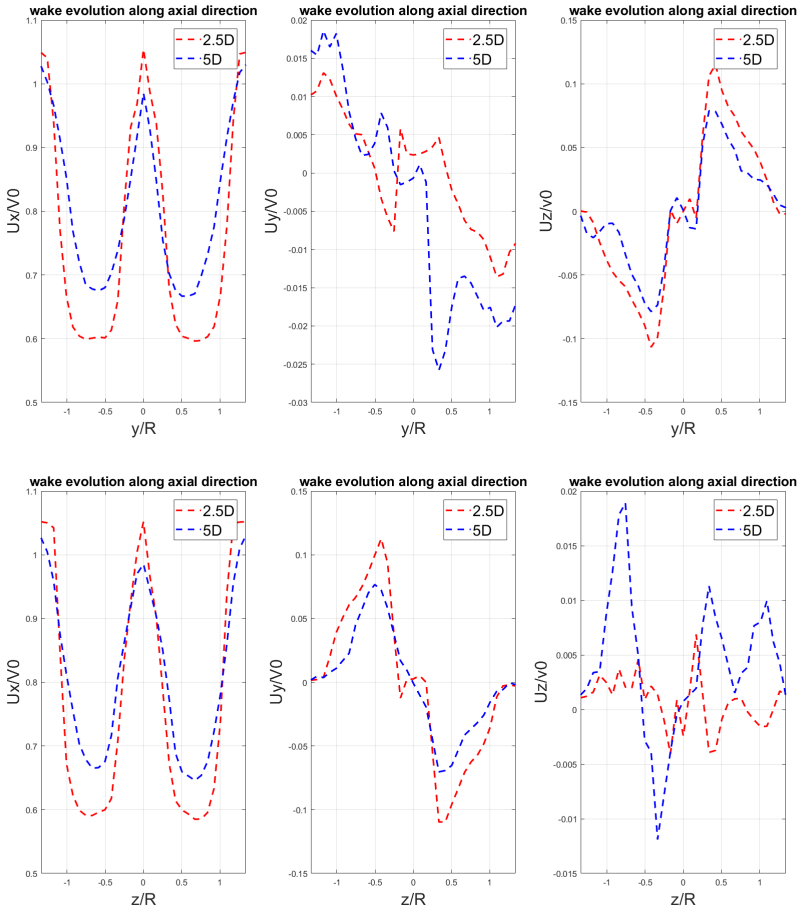


Figure C.6. Velocity profile evolution along axial direction measured both by horizontal (top) and vertical (bottom) transverse for p2.6

C.1.4 p2.8

Loads

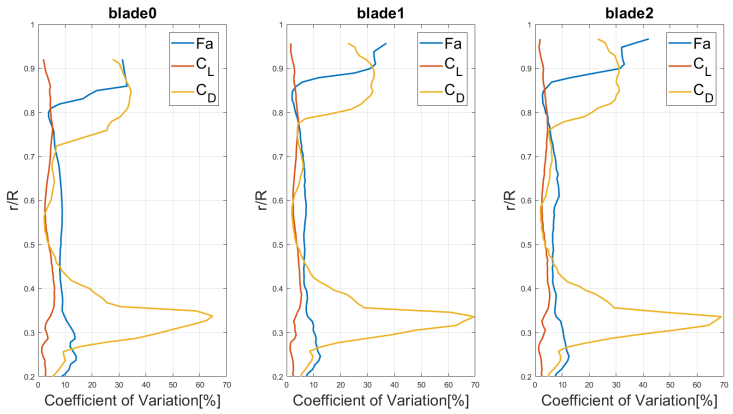


Figure C.7. Coefficient of Variation of loads distribution for p2.8

Wake Evolution

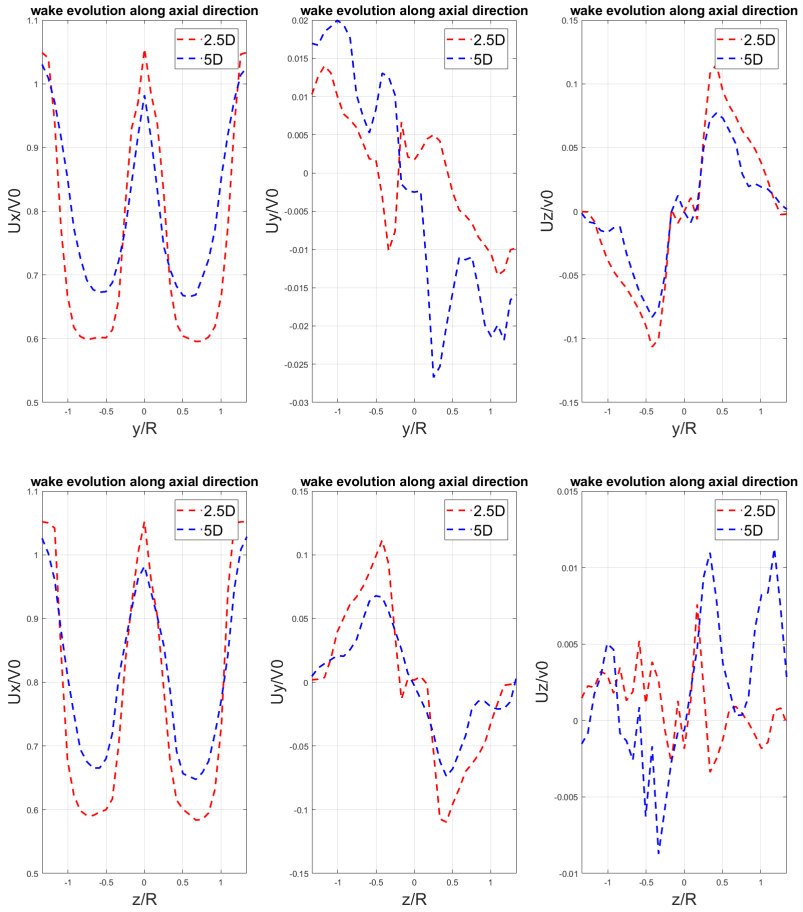


Figure C.8. Velocity profile evolution along axial direction measured both by horizontal (top) and vertical (bottom) transverse for p2.8

Acronyms

GVPM	Galleria del Vento Politecnico di Milano
FD	Finite Difference
FV	Finite Volume
OOP	object oriented programming
FOWT	Floating Offshore Wind Turbines
TSR	Tip Speed Ratio
CFD	Computational Fluid Dynamics
NS	Navier-Stokes
DNS	Direct Numerical Simulation
LES	Large Eddy Simulation
RANS	Reynolds Averaged Navier Stokes
DES	Detached Eddy Simulation
SGS	Subgrid-scale
PISO	Pressure-Implicit with Splitting of Operators
AL	Actuator Line
EVM	Effective Velocity Model

Bibliography

[1] GWEC, "Global wind report," 2021.

[2] T. Sebastian and M. A. Lackner, "Characterisation of the unsteady aerodynamics of offshore floating wind turbines," *Wind Energy*, 2013.

[3] T. Tran and D. H. Kim, "The aerodynamiinterference effects of a floating offshore wind turbine experiencing platform pitching and yawing motions," *Journal of Mechanical Science and Technology*, vol. 29, no. 2, pp. 549–561, 2015.

[4] ——, "Fully coupled aero-hydrodynamic analysis of a semi-submersible fowt using a dynamic fluid body interaction," *Renewable Energy*, vol. 92, pp. 244–261, 2016.

[5] A. Roberson, "Oc6 phase i: Investigating the underprediction of low-frequency hydrodynamic loads and responses of a floating wind turbine," *Journal of Physics*, vol. 1618, 2020.

[6] A. Robertson, R. Bergua, A. Fontanella, and J. Jonkman, "Oc6 phase iii definition document."

[7] I. Bayati, M. Belloli, L. Bernini, and A. Zasso, "Wind tunnel validation of aeroddyn with lifes50+ project: imposed surge and pitch tests." *Journal of Physics*, 2016.

[8] I. Bayati, "Unaflow project:unsteady aerodynamics of floating wind turbines," *Journal of Physics*, 2018.

[9] L. Bernini, "Unaflow: Report," Politecnico di Milano, Tech. Rep., 2018.

[10] I. Bayati, M. Belloli, L. Bernini, and A. Zasso, "Aerodynamic design methodology for wind tunnel tests of wind turbine rotors," *Journal of Wind Engineering and Industrial Aerodynamics*, vol. 167, pp. 217–227, 2017.

[11] I. Bayati, M. Belloli, A. Facchinetti, and A. Giappino, "Wind tunnel tests on floating offshore wind turbines proposal for hardware-in-the-loop approach to validate numerical codes," *Wind Engineering*, 2013.

[12] M. Sathyajith, *Wind Energy Fundamentals, Resource Analysis and Economics*, Springer, Ed., 2006.

[13] J. W. McCroskey, *The phenomenon of Dynamic Stall*, 1981.

[14] J. W. Larsen, S. R. K. NIelsen, and S. Krenk, "Dynamic stall model for wind turbine airfoils," *Journal of Fluids and Structure*, vol. 23, 2007.

[15] S. Mancini, "An experimental, analytical and numerical study of fowt's unsteady aerodynamics."

[16] S. Mancini, K. Boorsma, M. Caboni, M. Cormier, T. Lutz, P. Schito, and A. Zasso, "Characterization of the unsteady aerodynamic response of a floating offshore wind turbine to surge motion," *Wind Energy Science*, 2020.

[17] D. Boldrin, "Wind turbines unsteady aerodynamics: Experimental analysis of thrust and wake velocity."

[18] Y. Huang and W. Decheng, "Investigation of interference effects between wind turbine and spar-type floating platform under combined wind-wave excitation," *Sustainability*, 2019.

[19] T. Sebastian and M. A. Lackner, "Characterization of the unsteady aerodynamics of offshore floating wind turbines," *Wind Energy*, vol. 16, pp. 339–352, 2013.

[20] F. Berger, "Experimental investigation of dynamic inflow effects with a scaled wind turbine in a controlled wind tunnel environment," *Journal of Physics*, vol. 1037, 2018.

[21] B. Wen, "The power performance of an offshore floating wind turbine in platform pitching motion," *Energy*, vol. 158, pp. 508–521, 2018.

[22] ——, "Wind shear effect induced by the platform pitch motion of a spar-type floating wind turbine," *Renewable Energy*, vol. 135, pp. 1186–1199, 2019.

[23] G. Persico, A. Ortolani, J. Drofelnik, A. Jackson, and M. Campobasso, "Cross-comparative analysis of loads and power of pitching floating offshore wind turbine rotors using frequency-domain navier-stokes cfd and blade element momentum theory," *Journal of Physics*, vol. 1618, 2020.

[24] J. Wolkovitch, "Analytical prediction of vortex ring boundaries for helicopters in steep descents," *Journal of the American Helicopter Society*, 1972.

[25] J. Dong and A. Viré, "Comparative analysis of different criteria for the prediction of vortex ring state of floating offshore wind turbines," *Renewable Energy*, vol. 163, pp. 882–909, 2021.

[26] H. Versteeg and W. Malalasekera, *An Introduction to Computational Fluid Dynamics The Finite Volume Method.* Pearson, 2007.

[27] G. Montenegro and G. Persico, "Lecture notes of the course of modeling techniques for fluid machines."

[28] J. N. Sørensen and W. Z. Shen, "Numerical modelling of wind turbine wakes," *Journal of fluids engineering*, vol. 124, pp. 393–399, 2002.

[29] C. Bak, "Description of the dtu 10 mw reference wind turbine," *DTU Wind Energy*, 2013.

[30] I. Bayati, "Scale model technology for floating offshore wind turbines," *IET Renewable Power Generation*, pp. 112'–1126, 2017.

[31] ——, "Experimental investigation of the unsteady aerodynamics of fowt thrugh piv and hot-wire wake measurements," *Journal of Physics*, 2018.

[32] I. Bayati, M. Belloli, L. Bernini, and A. Zasso, "Wind tunnel wake measurements of floating offshore wind turbines," *Energy Procedia*, 2017.

[33] A. Fontanella, "Unaflow: a holistic experiment about the aerodynamics of floating wind turbines under imposed surge motion," *Wind Energy Science*, 2021.

[34] P. Schito, "Large eddy simulation of wind turbines: interaction with turbulent flow."

[35] J. N. Sørensen and K. O. Dag, "A new tip correction for actuator line computations," *Wind Energy*, vol. 23, pp. 148–160, 2020.

[36] A. R. Meyer Forsting, "A vortex-based tip/smearing correction for the actuator line," *Wind Energy Science*, vol. 4, pp. 369–383, 2019.

[37] J. N. Sørensen, "Simulation of wind turbines wakes using the actuator line technique," 2015.

[38] L. Bernini and M. M. Caccialanza, "Development of the effective velocity model for wind turbines aerodynamics numerical simulation through an actuator line approach."

[39] P. F. Melani, "Experimental assessment of an actuator line simulation tool for vertical axis wind turbines."

[40] H. Rahimi, "Evaluation of different methods for determining the angle of attack on wind turbines blades with cfd results under axial inflow conditions," *Renewable Energy*, vol. 125, pp. 866–876, 2018.

[41] M. Hansen, N. Sørensen, and y. Sørensen, J and Michelsen, J, "Extraction of lift, drag and angle of attack from computed 3d viscous flow around a rotating blade."

[42] I. Herreàez, E. Daniele, and J. Schepers, "Extraction of the wake induction and angle of attack on rotating wind turbine blades from piv and cfd," *Wind Energy Science*, vol. 3, pp. 1–9, 2018.

[43] M. J. Churchfiled, "An advanced actuator line method for wind energy applications and beyond," *AIAA SciTech Forum*, 2017.

[44] K. Boorsma and M. Caboni, "Numerical analysis and validation of unsteady aerodynamics for floating offshore wind turbines," TNO Institute, Tech. Rep., 2020.

[45] L. Martìnez-Tossas and C. Meneveau, "Filtered lifting line theory and application to the actuator line model," *Journal of Fluid mechanics*, vol. 863, pp. 269–292, 2019.