POLITECNICO DI MILANO

SCUOLA DI INGEGNERIA INDUSTRIALE E DELL'INFORMAZIONE

Master Degree Course in Aeronautical Engineering

Master Degree Thesis

A PRELIMINARY STUDY ON A WAKE MIXING TECHNIQUE FOR WIND FARM CONTROL

Supervisor
prof. Alessandro Croce

Co-Supervisor
prof. Stefano Cacciola

Candidate
Umberto Tettamanti
student ID: 863674

Anno accademico 2017–2018

Acknowledgements

Prima di esporre il lavoro compiuto in questa tesi di laurea magistrale vorrei ringraziare tutti coloro che hanno contribuito al raggiungimento di questo importante traguardo.

Ringrazio innanzitutto il profressor Alessandro Croce per aver reso possibile questa tesi e per avermi fornito gli stumenti e le conoscenze necessarie per portare a termine con successo il lavoro.

Un grosso ringraziamento va anche al mio correlatore Stefano Cacciola per il supporto e la pazienza avuta in questi mesi.

Ringrazio la mia famiglia per aver sempre creduto in me durante questo percorso al Politecnico di Milano. Ringrazio i miei genitori Paola e Maurizio per avermi sostenuto ogni giorno ed aver reso possibile il conseguimento di questo titolo di studio. Un grazie sincero va anche a mia sorella Gloria per aver sempre fatto il possible per aiutarmi.

Un ringraziamento speciale va alla mia ragazza Marta per avermi supportato e soprattutto sopportato durante questi anni universitari. È stato un periodo a volte complicato e senza il tuo sostegno ed il tuo continuo spronarmi a fare di meglio sicuramente non sarei stato in grado di raggiungere questo obbiettivo.

Ringrazio i miei compagni di studi, in particolare Giorgio e Mauro, per aver reso questo percorso di studi più divertente e meno difficoltoso.

Infine ringrazio i miei amici di sempre: Simone, Jordan, Xavier, Daniele, Emanuele, Domenico, Roberto, Fabio e Mirco. Grazie per tutti i bei momenti trascorsi insieme e per avermi permesso di staccare la spina quando ne avevo bisogno.

Contents

List of Tables VI

List of Figures VII

Summary IX

Sommario X

1 Introduction 1
1.1 Thesis outline . . . 2

2 State of the art 3

3 Virtual Environment 8
3.1 Governing equations . . . 8
3.2 Methodology . . . 11
3.2.1 Numerical methods . . . 11
3.2.2 Turbulence models . . . 12
3.2.3 Loads estimation . . . 13
3.2.4 Computational cost . . . 14
3.3 Case study . . . 15

4 Tools 16
4.1 Marconi HPC Cluster . . . 16
4.2 SOWFA . . . 17
4.3 FAST . . . 19
4.4 Setup . . . 21
4.5 Validation . . . 21

5 Models 24
5.1 Wind turbine properties . . . 24
5.2 Domain properties . . . 25
5.3 Wind properties . . . 26
5.4 Numerical parameters of the simulation . . . 27
5.5 Velocity Probes . . . 28

6 Simulations results 30
6.1 Effects of periodic collective motion on the wake . . . 30
6.2 Parametric Analysis . . . 37
6.3 Comparison between FAST-coupled SOWFA simulations and standalone FAST simulations . . . 40

6.4 Detailed results for optimal periodic collective motion 43
6.4.1 Effects on the flow field 43
6.4.2 Effects on power production 43
6.4.3 Mechanical effects on the wind turbine 46

7 Conclusions 55

A Installation of SOWFA and FAST on Marconi 57
A.1 Installation of OpenFOAM 2.4.x 57
A.2 Installation of FAST v8 59
A.3 Installation of SOWFA 60

B Loads on the wind turbine rotor used for the computation of DELs 62

C Effects of an oscillating flow field on the power production 65

Bibliography 68

List of Tables

4.1 Comparison between MARCONI and NREL HPC performance. 23
5.1 NREL off-shore 5MW baseline wind turbine – Main properties. 24
6.1 Average wake velocity – with reference wind velocity of 6.5 m/s. 43
6.2 Average wake velocity – with reference wind velocity of 9 m/s. 44
6.3 Power production of the two turbines system – reference wind velocity of 6.5 m/s. 46
6.4 Power production of the two turbines system – reference wind velocity of 9 m/s. 46
6.5 Blade root forces damage equivalent loads. 50
6.6 Hub damage equivalent loads. 50
6.7 Normalized flapwise damage equivalent loads – relative increases. 52
6.8 Normalized thrust damage equivalent loads – relative increases. 52
6.9 PCM activation regions. 53
6.10 Lifetime equivalent impact of PCM control at 6.5 m/s wind speed on blade root loads. 54
6.11 Lifetime equivalent impact of PCM control at 6.5 m/s wind speed on hub thrust force. 54

List of Figures

2.1 Simulation of downstream wakes on a wind farm. Credit: Kenny Gruchalla, NREL. Picture from [1] . . . 3
2.2 Normalized power deficit for a reference wind speed of V=8±0.5 m/s. Picture from [2]. . . . 4
2.3 Power production, averaged over the three turbines per row. "HF" denotes the name of the axial induction control strategy used by Corten and Schaak. Picture from [3]. . . . 5
2.4 Comparison of YM and CyPC in terms of wake velocity (a) and wake redirection (b). Picture from [4]. . . . 6
2.5 Snapshots of the flow field at different Strouhal numbers. Left: isocontours of vorticity, right: streamwise velocity. Picture from [5]. . . . 7
2.6 Left: wind farm power extraction increase, right: row-averaged power extraction. Picture from [5]. . . . 7
3.1 Centerline velocity of a turbulent jet. Picture from [6]. . . . 9
3.2 Energy cascade on logarithmic axis with eddies length scales. Picture from [6]. 10
3.3 Kolmogorov model for energy spectrum function. . . . 11
3.4 One-dimensional spectra from experimental data and model spectrum (continuous lines). Picture from [6]. . . . 11
3.5 Actuator Line Model of a wind turbine rotor. . . . 14
4.1 FAST v8 block scheme for floating off-shore wind turbines. . . . 20
4.2 Shaft coordinate system. Picture from [7]. . . . 20
4.3 Blade coordinate system. Picture from [7]. . . . 20
4.4 ABL flow field generated by our simulation. . . . 22
4.5 ABL flow field from NREL simulation [8]. . . . 22
4.6 Simulation outputs with the controller compiled on MARCONI. . . . 23
4.7 Simulation outputs with the provided controller. . . . 23
5.1 Controller regulation trajectory – Pitch angle, rotation speed, torque. . . . 25
5.2 Controller regulation trajectory – Nominal power curve. . . . 25
5.3 Domain layout. . . . 26
5.4 Refinement Regions in the x-z and y-z planes. . . . 27
5.5 Numerical mesh – y-z plane view. . . . 28
5.6 Numerical mesh – close-up view of the refined region. . . . 28
5.7 Sampling probes setup. . . . 28
6.1 Effects of PCM on blade pitch angle, rotor speed and power production. . . . 30
6.2 Instantaneous flow velocity at 3, 4 and 5 diameters – without PCM (left) and with PCM control (right). . . . 32
6.3 Instantaneous flow velocity at 6, 7 and 8 diameters – without PCM (left) and with PCM control (right). . . . 33
6.4 Average flow velocity U_{ave} at 3, 4 and 5 diameters – without PCM (left) and with PCM control (right). . . . 34

6.5 Average flow velocity U_{ave} at 6, 7 and 8 diameters – without PCM (left) and with PCM control (right). 35
6.6 Velocity profile – red dashed line: without PCM, blue line: with PCM. . . . 36
6.7 Velocity profile at 3 diameters – measured values and interpolated curve. . . 36
6.8 Velocity profile at 6 diameters – measured values and interpolated curve. . . 36
6.9 Velocity profile – values for different frequencies ω_{PCM}. 37
6.10 Velocity profile at 5 and 6 diameters – values for different frequencies ω_{PCM}. 38
6.11 Rotor-averaged instantaneous velocity – values for different frequencies ω_{PCM}. 38
6.12 Normalized average wake velocity – values for different frequencies ω_{PCM} and amplitudes θ_{PCM}. 39
6.13 Comparison of different simulations in the same trim point – Generated power. 41
6.14 Comparison of different simulations in the same trim point – Rotor thrust. . 41
6.15 Comparison of standalone FAST and SOWFA simulations with PCM – Generated power. 41
6.16 Comparison of standalone FAST and SOWFA simulations with PCM – Rotor thrust. 41
6.17 Comparison of standalone FAST and SOWFA simulations with PCM – Spectrum of generated power. 42
6.18 Average flow velocity at 4 diameters – with optimal PCM and without PCM. 43
6.19 Average flow velocity at 5 diameters – with optimal PCM and without PCM. 44
6.20 Average flow velocity at 6 diameters – with optimal PCM and without PCM. 44
6.21 Rotor-averaged instantaneous velocity at 6 diameters – values for optimal PCM parameters. 45
6.22 Thrust force on the rotor. 46
6.23 Torque on the rotor (upper) and on the generator (lower). 47
6.24 Rotor power (upper) and generator power (lower). 48
6.25 Power (upper), torque (middle) and thrust (lower) coefficients. 49
6.26 Normalized flapwise damage equivalent loads – bar chart. 51
6.27 Normalized thrust damage equivalent loads – bar chart. 51
6.28 $FINO3$ wind rose. 52
6.29 $FINO3$ wind speed distribution. 52
6.30 Wind farm layout. 53
B.1 Blade root forces – without PCM control. 63
B.2 Blade root forces – with PCM control. 63
B.3 Hub forces – without PCM control. 64
B.4 Hub forces – with PCM control. 64
C.5 Oscillating velocity (upper), average value function property (lower). 66

Summary

Wind farm control is a very actual and appealing topic in the wind energy community: its benefits are a higher overall power production and a longer lifetime of the wind turbines, with a consequent reduction of the cost of energy. Many solutions have been proposed and studied in the past few years but yet have to be implemented on large scale in commercial wind farms. This thesis will focus on the most recently proposed solutions that aim at enhancing wake mixing to guarantee a faster recovery of the turbine wake. We propose a control strategy that aims at enhancing the wake mixing through a periodic collective motion of the blades, in particular a sinusoidal variation of the blade pitch angle. The enhancement of the wake mixing will be obtained thanks to the resulting cyclic variation of the thrust force on the rotor. The methodology used to study this problem is based on numerical simulations of the flow field. The first aim of this thesis is therefore to provide to the POLI-Wind research group the necessary means for performing high resolution aerodynamic numerical simulations of wind farms. The second aim is to study the effectiveness of the wake mixing enhancement technique using the aforementioned tools. The tools chosen are SOWFA and FAST, both developed by NREL for CFD simulations and wind turbine structural analysis. The effectiveness of the periodic collective motion control technique is evaluated not only in terms of wake velocity but also considering the loads induced on the blades and how they impact on the wind turbine lifetime. To measure the incidence of the loads on the turbine lifetime we decided to compute the Damage Equivalent Load caused by the activation of the wake mixing technique. According to the very preliminary results obtained it is possible to conclude that the periodic collective motion control technique can be successfully implemented to increase the overall power production of a wind farm. On the other hand, we observed a sensible increase in the damage equivalent loads on the blades. Therefore, the proposed control strategy seems effective, but its implementation must be evaluated carefully to avoid excessive loadings on the wind turbine.

Sommario

Il controllo di wind farm è un argomento di grande attualità nella comunità eolica visti i notevoli benefici che porterebbe: in primis l'aumento della produzione energetica del campo eolico, ma anche un miglioramento della vita utile dell'impianto e la conseguente riduzione del costo dell'energia. Nel corso degli ultimi anni sono state proposte e studiate numerose soluzioni, ma al momento nessuna è ancora stata implementata in campi eolici commerciali. Questa tesi si concentra su una tipologia di soluzioni proposta recentemente e basata sull'idea di aumentare il rimescolamento delle scie delle turbine in modo tale da garantire un migliore flusso a valle. Abbiamo proposto una tecnica per l'aumento del rimescolamento basata sul moto periodico e collettivo delle pale, in particolare imponendo una variazione sinusoidale dell'angolo di pitch. Ne consegue una variazione ciclica della spinta sul rotore che è il meccanismo alla base dell'aumento di rimescolamento. La metodologia scelta per affrontare questo problema è basata su simulazioni numeriche del campo di moto. Il primo scopo di questa tesi è quindi dotare il gruppo di ricerca POLI-Wind degli strumenti necessari all'esecuzione di simulazioni CFD di campi eolici. Il secondo scopo è studiare l'efficacia della tecnica proposta sfruttando i suddetti strumenti. Gli strumenti selezionati sono SOWFA e FAST, due software sviluppati dal NREL rispettivamente per simulazioni CFD e analisi strutturale di turbine eoliche. L'efficacia della tecnica di controllo basata su moto periodico collettivo è stata valutata non solo in termini di velocità della scia, ma anche in termini di carichi indotti sulle pale, stimando l'impatto che possono avere sulla vita utile della turbina eolica. Per stimare l'incidenza dei carichi generati sulla vita della turbina abbiamo deciso di calcolare il Damage Equivalent Load provocato dall'attivazione della strategia di controllo per aumento del rimescolamento. Dai risultati di quest'analisi preliminare è possibile concludere che la tecnica di controllo basata su moto periodico collettivo può essere efficacemente impiegata per aumentare la produzione energetica di un campo eolico. Dal punto di vista dei carichi si è però osservato un incremento sensibile nei Damage Equivalent Loads sulle pale. Pertanto la strategia di controllo proposta sembra essere efficace, ma la sua implementazione deve essere valutata con attenzione, in modo da non incorrere in carichi eccessivi sulla turbina eolica.

Chapter 1

Introduction

Nowadays wind farm control is a very actual and appealing topic in the wind energy community: many possible solutions have been proposed and analysed in the past few year, but yet have to be implemented on large scale in commercial wind farms.

In particular, great interest is directed to the analysis of wakes shed by upstream wind turbines since this is the main form of interaction between different machines in the wind farm.

The most recently proposed solutions aim at enhancing wake mixing in order to guarantee a faster recovery of the turbine wake and hence less induced disturbances in the downstream machines. Wind tunnel tests and numerical simulations are fundamental to study this kind of solutions.

A first aim of this thesis is to provide to the POLI-Wind research group the necessary means for performing high resolution aerodynamic numerical simulations of wind farms. This would allow the group to perform a wider range of analysis, taking into account virtually every aspect of wind turbines and wind farms. The tools chosen are SOWFA and FAST, both developed by NREL for CFD simulations and wind turbine structural analysis.

The second aim of thesis is evaluating the effectiveness of a wake mixing enhancement technique using the aforementioned tools.

In this thesis we proposed a control strategy that aims at enhancing the wake mixing through a periodic collective motion of the blades. Munters and Meyers in [5] showed that a cyclic variation of the thrust coefficient can exploit the wake meandering phenomenon to enhance the wake mixing. We used a more accurate model for the wind turbine rotor (the Actuator Line Model instead of the Actuator Disc Model used in [5]) and we imposed a sinusoidal variation of the collective pitch angle. The variation of the collective pitch angle has been tuned in order to obtain a cyclic variation of the thrust coefficient equivalent to the one proposed in [5]. The thrust coefficient based approach used by Munters and Meyers may look more straightforward, but it has the major drawback that the thrust coefficient is not a quantity that can be easily estimated or measured by sensors. On the other hand we proposed a more realistic and practical approach, easier to implement into the control system since the blade pitch angle is already used as a input/output parameter by virtually all the modern wind turbine control systems.

The effectiveness of the periodic collective motion control technique is evaluated not just in terms of wake velocity but also considering the loads induced on the blades and the wind turbine structure. So far the wake recovery techniques have been studied in terms of effects produced on the flow field, typically measuring the wake velocity deficit, and the performance is expressed only in terms of increase in the overall power production of the wind farm. The behaviour of the wind turbine structure has often been neglected, for example the use of an Actuator Disc Model in [5] de facto ignores completely the real rotor geometry. We wanted to

perform a wider analysis, taking into account also the cost of the periodic collective motion control technique in terms of load increment and impact on the wind turbine lifetime.

Our objective is to understand whether the implementation of a wake mixing technique, like the periodic collective motion, is effective regardless of the increase in the loads on the structure or if this increase has a non-negligible cost that might affect the choice of a wind farm control technique. We expect that a possible optimal solution might be a trade-off between wake mixing performance and mitigation of loading increase.

1.1 Thesis outline

To present this work we structured the thesis in seven chapters:

Chapter 1 is an introduction to the problem tackled in this thesis. It includes a description of the aims of the thesis as well as a focus on its innovative content.

Chapter 2 is an overview of the context of this thesis with a detailed definition of the state of the art of wind farm control techniques and a brief introduction to the european H2020 CL-Windcon project .

Chapter 3 is a description of the virtual environment required for performing the desired analysis. Furthermore there is an introduction to the periodic collective motion case study.

Chapter 4 is a focus on the software chosen for performing the simulation and the reasons behind this particular choice.

Chapter 5 is a detailed description of the models implemented in the numerical simulations: the wind turbine and its control system are characterized, the simulation domain and the numerical mesh are defined, the wind that is imposed and the simplifications made.

Chapter 6 reports the outcomes of the simulations. Firstly it presents an analysis of the behaviour of the wake in the case study. Then a parametric analysis on the control technique parameters is performed, with the aim of optimizing the wake mixing. The last section reports detailed results in terms of wake velocity, power production, and structural loadings for the optimal values of the control technique parameters.

Chapter 7 draws the conclusions from the work done in this thesis and presents possible future developments.

Chapter 2

State of the art

Nowadays large wind farms with multiple wind turbines are becoming the most common way to produce wind energy. In spite of this fact, the currently adopted control techniques aim at the maximization of power production of each single turbine, without taking into account any kind of interaction between the turbines. The interactions within a wind farm are mainly due to the wakes originating from the upwind turbines (see Fig. 2.1). The wake of a wind turbine is a turbulent flow region characterised by a deficit of kinetic energy and complex vortical structures. The interaction between wakes and other wind turbines causes two major issues:

- a deficit in the effective wind speed seen by the downstream turbines, which translates into a reduced power production.
- an increase in the loads on turbines in the wake region, due to higher turbulence and velocity shear

To give an idea on the relevance of this issues, reductions up to 40 % in energy extraction, compared to isolated turbines, have been measured in operational off-shore wind farms [2]. Figure 2.2 shows the power production measured in the Horns Rev off-shore wind farm. This is why in recent years there is a growing interest in control strategies acting at wind

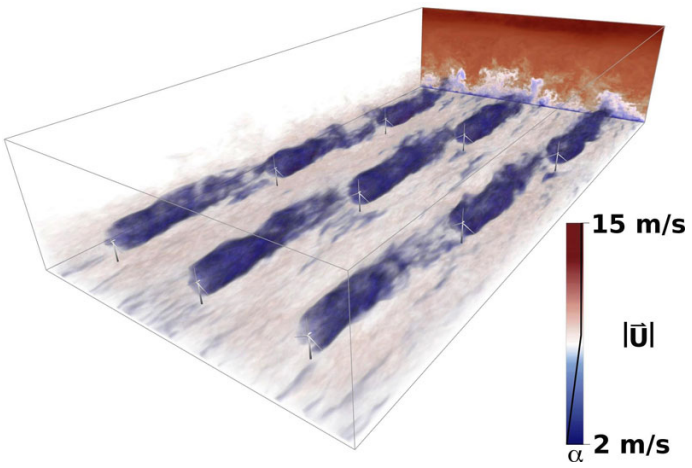


Figure 2.1: Simulation of downstream wakes on a wind farm. Credit: Kenny Gruchalla, NREL. Picture from [1]

.

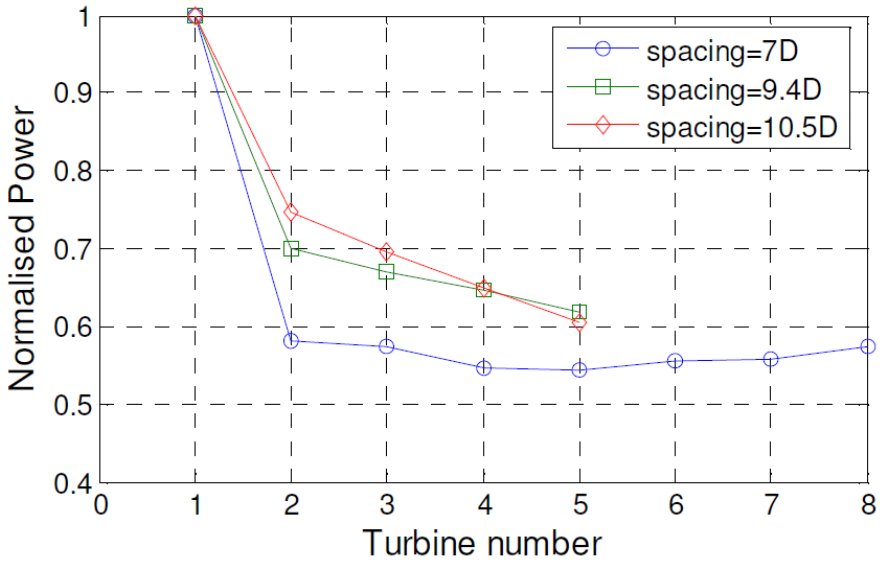


Figure 2.2: Normalized power deficit for a reference wind speed of V=8±0.5 m/s. Picture from [2].

farm level (sometimes referred to as *Super-Controllers*) that aim at maximizing the overall plant production instead of the single turbines' one. The best way to achieve this goal is still unclear, but many promising solutions have been proposed and studied.Their practical implementation is currently being evaluated.

As anticipated above, wind farm control is a new topic that emerged only in recent years. Solutions proposed earlier have been studied in an extensive way and a large body of literature is available, with numerical simulations as well as wind tunnel tests. See [9, 10, 11] and references therein. On the other hand latest solutions are still under study and only preliminary results have been published.

Generally speaking we can identify three main families of wind farm control techniques:

Axial induction control. This control strategy is based on the idea of reducing the downstream velocity deficit by deviating the turbine operative status from the optimal one. Operating the turbine at sub-optimal tip speed ratios and blade pitch angles will cause a reduction in the power production but at the same time it will produce a weaker wake, so one hopes that the power production loss will be recovered thanks to a higher power production of the downstream turbines. This approach has been first proposed in 1998 when Steinbuch et al. [12] theorized that the power output of a wind farm could be enhanced by reducing the tip speed ratio of upwind turbines. Later in 2003 Corten and Schaak [3] showed with data from wind tunnel testing that the overall production of a 6 by 3 wind farm could be increased up to 4.5% by decreasing the axial induction factor below the optimal value of $\frac{1}{3}$. The results obtained by Corten and Schaak are reported in Fig. 2.3.

Other studies also predicted overall energy production gains in the order of 5%, nevertheless more recent works like [13] showed that in many circumstances axial induction control does not produce significant gains.

New approaches like the adaptive axial induction control proposed by Gebraad and van Wingerden in [14] or the dynamically integrated axial induction control proposed by Munters and Meyers in [15, 16] seem to be a more effective way to implement the

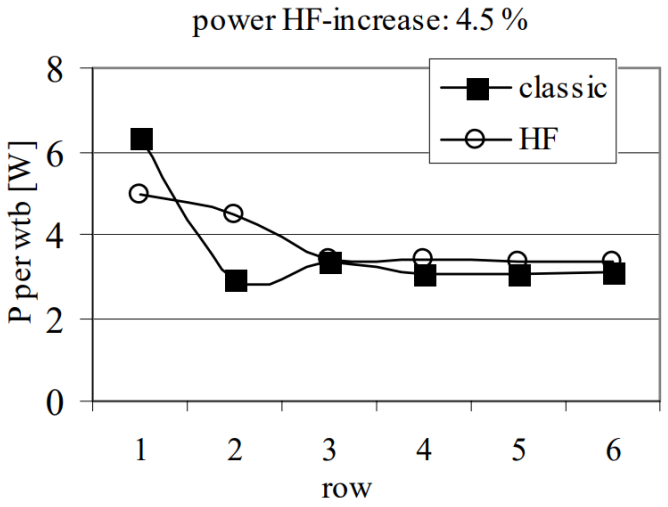


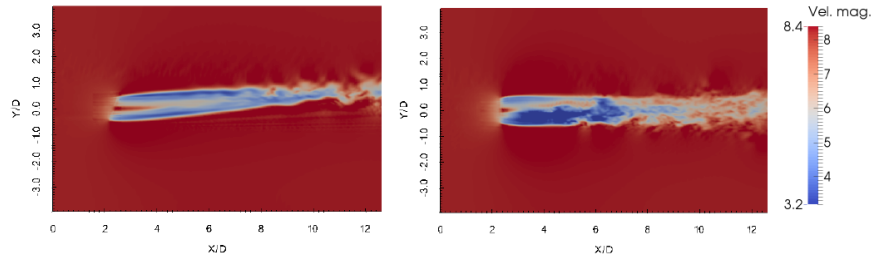
Figure 2.3: Power production, averaged over the three turbines per row. "HF" denotes the name of the axial induction control strategy used by Corten and Schaak. Picture from [3].

axial induction control. In adaptive axial induction control interactions are estimated through simplified wake models and taken into account by the controller. In dynamically integrated control the axial induction control is combined with other wind farm control techniques and activated only for particular wind conditions.

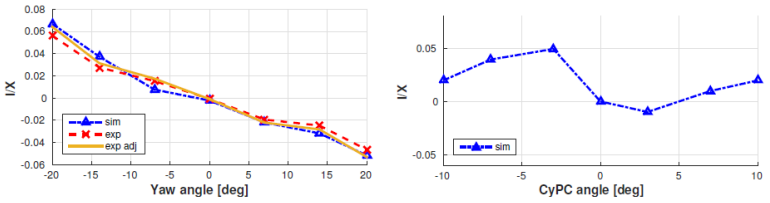
Wake redirection. The idea behind this kind of control strategy is to reduce the exposure of downwind turbines to the wakes shed by upstream turbines by misaligning the latter with respect to the wind direction. In practice the misalignment can be achieved in two ways: with a yaw maneuver or with a tilt maneuver of the nacelle.
Apart from turbine misalignment, a method to obtain the same wake redirection by a cyclic variation of the blades' pitch angle has been recently studied. Individual pitch and cyclic pitch control strategies have already been studied in the field of load reduction [17, 18] with positive results. Wang et al. noted a redirecting of the wake as side effect of the cyclic pitch control (CyPC) and they studied the phenomenon [4]. They varied the pitch angle of each blade according to:

$$\theta_i = \theta_0 + \theta_c \cos(\psi_i + \gamma) \qquad \text{where} \quad \begin{cases} \theta_0 \text{ is the collective pitch constant} \\ \theta_c \text{ is the 1P pitch amplitude} \\ \psi_i \text{ is the azimuthal angle of the i blade} \\ \gamma \text{ is the phase angle} \end{cases} \tag{2.1}$$

To estimate the effectiveness of CyPC, they performed a comparison with yaw misalignment (YM) control using numerical simulations and test in Politecnico di Milano wind tunnel using scaled models. The results in terms of wave velocity and wake displacement are reported in Fig. 2.4. Yaw misalignment results to be more effective than CyPC in terms of lateral displacement of the wake, because it can generate higher lateral forces. In particular the lateral displacement shows a linear dependency from the misalignment angle and it is independent from other operative parameters, so Wang et al. concluded that YM is expected to produce similar effects on other wind turbines. Another paper from Gebraad et al. [2] confirms the effectiveness of yaw misalignment



a): Velocity magnitude $|\vec{u}|$ for YM (left) and CyPC (right).



b): Wake displacement for YM vs. yaw misalignment φ (left), and for CyPC vs. cyclic pitch amplitude θ_c (right).

Figure 2.4: Comparison of YM and CyPC in terms of wake velocity (a) and wake redirection (b). Picture from [4].

control technique and estimates plant energy production increases up to 15%.
Yaw misalignment appears to be the most promising control technique: it is the most effective in redirecting the wake and since virtually all wind farm turbines already have yaw motion actuators it should be easy to implement.

Fast wake recovery This family of control techniques is the most recent one since it has emerged only in the past few years and many of its aspects still have to be investigated. It is based on the idea that the low-frequency periodic wake displacement, known also as wake meandering, can be exploited to enhance the wake mixing. It has been shown by Medici et al. [19] that wake meandering depends on the number of blades, the rotation speed, the pitch angle and the thrust on the wind turbine. So by acting on the pitch angle or on the thrust coefficient one can modify the behaviour of the wake. Medici et al. in particular observed that wake meandering happened at a rather distinct frequency identified using the Strouhal number. The principle behind these techniques is that matching this particular Strouhal number during the operative cycle of a wind turbine should exploit the meandering mechanism and hence produce a faster wake recovery. Munters and Meyers in [5] followed this principle to optimize the power production of 4×4 wind farm. Their simulations confirmed that there is an optimal Strouhal number that enhances the wake mixing. Flow visualizations for different Strouhal numbers are reported in Fig. 2.5. Munters and Meyers also measured an increase in the power production of the wind farm up to + 6 %, the results are reported in Fig. 2.6.

Even if this technique aims at the same external effects of the axial induction control techniques, namely reducing the downstream wind velocity, it has the major advantage of avoiding the downrating of the turbine, which instead continues to work in a neighbourhood of the optimal conditions with small power production losses.

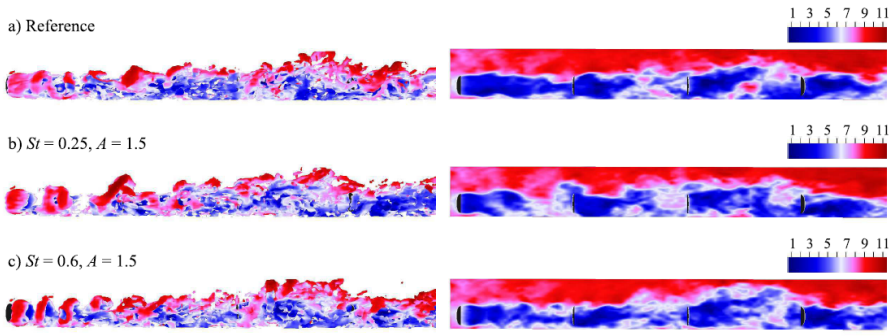


Figure 2.5: Snapshots of the flow field at different Strouhal numbers. Left: isocontours of vorticity, right: streamwise velocity. Picture from [5].

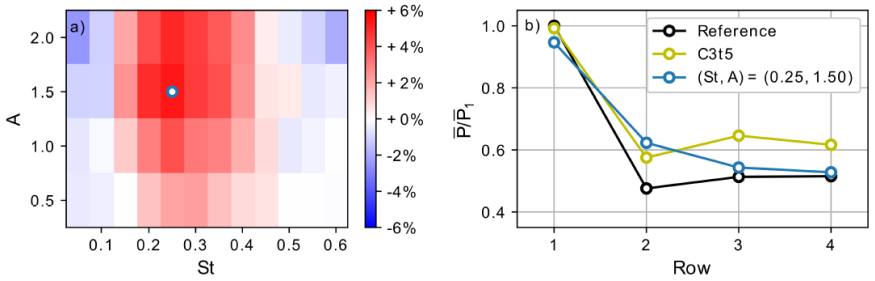


Figure 2.6: Left: wind farm power extraction increase, right: row-averaged power extraction. Picture from [5].

To further underline the relevance of wind farm control, it is worth citing the CL-Windcon project, part of the European Union's Horizon 2020 programme. CL-Windcon has arranged an international consortium of universities, including Politecnico di Milano, reserch centers and industrial companies. The aim of CL-Windcon project is developing closed-loop advanced control algorithms that treat the entire wind farm as a comprehensive real-time optimization problem. The success of this project will bring major benefits such as:

- higher energy production;
- improvement of the reliability;
- higher availability of the wind turbines;

It is expected that the project will reduce the cost of operation and maintenance of the wind plant, as well as improve the reliability and the availability of the plant.

All of this will translate into a lower cost of energy, which will help reaching the target of sustainability fixed by the Horizon 2020 programme.

Chapter 3

Virtual Environment

The physics behind the problem tackled in this thesis is complex: wind turbines are already complex machines and the fact that they are surrounded by a turbulent atmospheric boundary layer makes it impossible to find an analytical solution of the flow field. In this chapter we will present the governing equations of the problem and the tools that will be necessary for computing a numerical solution.

3.1 Governing equations

The governing principles for the flow field that interacts with the wind turbines are the conservation laws for mass, momentum and energy, which are also called the Navier-Stokes equations. The equations will be stated in differential form using the Eulerian description of the motion. In the Eulerian description the flow field's characteristics are monitored at fixed locations of the space, hence they depend on the spatial coordinate $\mathbf{x}$ and on time t.

Continuity equation

It is based on the principle that mass is neither created or destroyed: if we consider a volume that contains a fixed number of fluid particles, without sources of mass, it can stretch and move with the flow, but it will always contain the same fluid particles. This property is expressed by the equation:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{U}) = 0 \tag{3.1}$$

Where ρ is the fluid density and $\mathbf{U}$ is the velocity vector.

Navier-Stokes momentum equation

This equation is the equivalent of Newton's second law: the internal momentum is balanced by external forces. Here there is a further assumption that the fluid is Newtonian and hence the internal stresses are given by the pressure field and by a viscous term proportional to velocity.

$$\frac{\partial \rho \mathbf{U}}{\partial t} + \nabla \cdot (\rho \mathbf{U}\mathbf{U}) = -\nabla P + \mu \nabla^2 \mathbf{U} + \mathbf{f} \tag{3.2}$$

Where P is the pressure, μ is the dynamic viscosity and $\mathbf{f}$ are the external forces, like the ones imposed by the wind turbine.

Conservation of energy

This equation expresses the conservation of internal energy e, but alternative formulations using different quantities are possible and they might be a better option depending

on the particular physical problem.

$$\frac{\partial \rho e}{\partial t} + \nabla \cdot (\rho e \mathbf{U}) = -\nabla \cdot \mathbf{q}_F + \dot{q} + \nabla \cdot (\sigma \mathbf{U}) \tag{3.3}$$

Where $\mathbf{q}_F$ it the heat flux vector, $\dot{q}$ is an external heat source and σ is the stress tensor.

Under certain hypothesis some terms of the equations can be simplified, but we will focus on the assumptions made in the following chapters.

The Navier-Stokes equations apply equally to laminar and turbulent flows, but in this thesis we are interested only in the latter. When dealing with turbulence the flow field $\mathbf{U}$ is a random variable, in the sense that it does not have a unique value but it is characterized by a probability density function [6]: if we repeat an experiment under the same conditions at every repetition we will find a different value of $\mathbf{U}$ in a given location $\mathbf{x}$ at a given time t. It follows that the velocity field varies in a significant way in both position and time, as it is possible to see in Fig. 3.1.

However the turbulent motions follow a physical process called *energy cascade*, that was firt introduced by Richarson in 1922 and further theorized by Kolmgorov. The idea of energy cascade is that the turbulent motions range in size from large scales (also called eddies) to smaller ones. The kinetic energy is introduced in the flow at large scales level and is then transferred to smaller and smaller scales, where it is dissipated by viscous forces. In particular for a fully turbulent flow at high Reynolds number with characteristic length scale $\mathcal{L}$ and velocity $\mathcal{U}$:

Large eddies, also referred to as integral scales, have
length scale $\ell_0 \simeq \mathcal{L}$
characteristic velocity $u_0 = u(\ell_0) \simeq \mathcal{U}$
Reynolds number $Re = \frac{u_0 \ell_0}{\nu}$ is large

Small eddies, also referred to as Kolmogorov microscales, have
length scale $\eta \ll \ell_0$
characteristic velocity $v \ll U_0$
Reynolds number $Re = \frac{v\eta}{\nu} \sim 1$

The energy cascade process is sketched in Fig. 3.2.

It is evident that there is a range of scales that are large compared to η but small compared to l_0. This is the so called *inertial subrange* because, according to the Kolmogorov's second

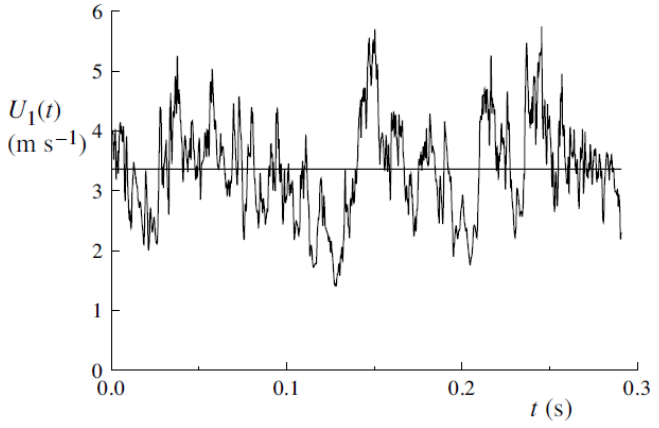


Figure 3.1: Centerline velocity of a turbulent jet. Picture from [6].

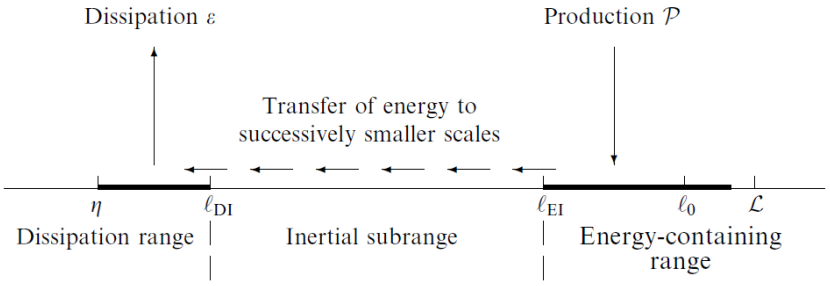


Figure 3.2: Energy cascade on logarithmic axis with eddies length scales. Picture from [6].

similarity hypothesis, the motion of this eddies is determined by inertial forces, since they are too large to be affected by viscosity.

The energy cascade process can be described in a more accurate way by looking at how turbulent kinetic energy is distributed among the various length scales. This can easily be done by analysing the energy spectrum $E(k)$, that is obtained from the Fourier transform of the two-point correlation of the flow field. Note that k is the wavenumber, defined as:

$$k = \frac{2\pi}{\ell} \quad \text{where } \ell \text{ is the length scale.}$$

Kolmogorov formulated a model for the energy spectrum in homogeneous isotropic turbulence:

$$E(k) = \underbrace{C\varepsilon^{2/3}k^{-5/3}}_{\mathbf{I}}\underbrace{f_L(k\mathcal{L})}_{\mathbf{II}}\underbrace{f_\eta(k\eta)}_{\mathbf{III}} \tag{3.4}$$

Where:

I is Kolmogorov's $-\frac{5}{3}$ spectrum that predicts the energy spectrum function in the inertial subrange.

II is a non-dimensional function that fixes the $-\frac{5}{3}$ spectrum in the energy-containing range, hence $F_L \sim 1$ when k is large.

III is a non-dimensional function that fixes the $-\frac{5}{3}$ spectrum in the dissipation range, hence $F_\eta \sim 1$ when k is small.

The model spectrum and spectra obtained from measurements are reported in Fig. 3.3 and 3.4.

From the plots in Fig. 3.4 we can note a crucial property of the energy spectrum: in the dissipation range it shows a self-similar behaviour, which means that for high wavenumbers the spectrum is universal. The importance of this property will be explained in the following chapters.

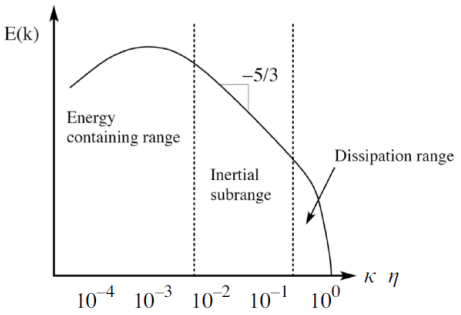


Figure 3.3: Kolmogorov model for energy spectrum function.

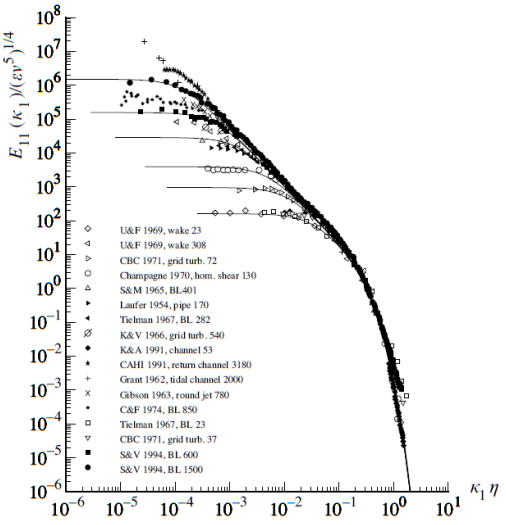


Figure 3.4: One-dimensional spectra from experimental data and model spectrum (continuous lines). Picture from [6].

3.2 Methodology

The Navier-Stokes equation can not be solved analytically, so we need to use CFD (Computational Fluid Dynamics) methods to solve them numerically. Nowadays CFD tools are used in many fields of engineering interest, like turbulence or heat transfer problems, producing accurate results at relatively low cost in terms of time and resources. CFD analysis follow a common procedure:

1. **Pre-processing**: this is the first step and it involves the creation of the geometry, the generation of the numerical mesh and the definition of suitable initial and boundary conditions.

2. **Simulation**: during this step the governing equation are solved numerically. According to the particular case, assumptions can be made to simplify the equations and reduce the computational cost. Different methods can be used to solve the equations, the most common are Finite Difference, Finite Element and Finite Volume.

3. **Post-processing**: this is the final steps where the outcomes of the simulations are analysed and often used to produce a visualization of the solution.

3.2.1 Numerical methods

Focusing on numerical methods for the solution of the equations, the three methods cited earlier are here briefly described:

Finite Difference Methods are based on the idea of approximating the derivatives in the equations as a difference between the values of the function on the nodes. This methods are typically used to solve Ordinary Differential Equations.

Finite Element Methods are typically used in structural analysis but can also be used for fluid dynamics. In FEM the differential equations are formulated in weak form and solved to find the solution in the mesh nodes.

Finite Volumes methods are the most common methods used in CFD codes. In FVM the domain is divided in cells and the differential equations are formulated in conservative form. The surface integrals in the equations are evaluated as surface fluxes at cell interfaces. Since the flux entering a given cell is identical to that leaving the adjacent one, FVM are conservative methods. Other advantages of FVM are the fact that they can be easily used with unstructured mesh and their higher performance in terms of computational time and required memory.

Since we are dealing with the complex geometry of wind turbine, the possibility to use unstructured meshes makes FVM the most suitable choice. Furthermore we want to simulate large wind farms for long time intervals and the higher computational efficiency of FVM is another key aspect. For this reasons we decided to use a CFD code based on Finite Volume methods.

3.2.2 Turbulence models

Now that we have chosen the method for solving the governing equations, it is important to focus on turbulence and the way it can be treated in numerical simulations. As explained in the previous section, turbulence is a very complex phenomena. The solution of all the length scales involved in turbulence requires a mesh resolution that implies prohibitive computational costs. In fact, it is very common to introduce models to reduce the computational cost of the simulations.

There are three main approaches to turbulence modelling:

Reynolds-averaged Navier–Stokes (RANS).
This is probably the oldest approach to turbulence modelling. It is based on the *Reynolds decomposition* operation that splits the variables into a time-averaged part and a fluctuating part. This operation produces a new set of equations that contains a new term called Reynolds stress tensor. A model for the Reynolds stress tensor is required to close the Reynolds averaged Navier-Stokes equations and different choices are available. There are two main choices for the model:

- **Eddy viscosity models**, based on the Boussinesq hypothesis, define the Reynolds stress tensor through one or more equations that depend on the eddy viscosity. Famous models like $k-\epsilon$, $k-\omega$ and Spalart-Allmaras models belong to this family.
- **Reynolds strees tensor models** (RSTM) model directly the tensor by adding a new partial differential equation to the problem.

The main advantages of RANS are the ease of use and the relatively low computational cost. On the other hand the decomposition operation causes a loss of information, like velocity PDF or turbulent structures. Moreover the accuracy of the solution strongly depends on the choice of a suitable turbulence model for the considered problem.

Large Eddy Simulations (LES).
As the name suggests this approach solves directly the large scale turbulent motions and introduces a model just for the small scale motions. As we anticipated in the previous section, the small scale motions have (to some extent) a universal character and hence it does make sense to remove them and introduce a model. Not to mention the fact that the direct solution of small scales is computationally very expensive, more than the large scales one. From the operative point of view, the small scales (often called subgrid scales) are removed through a filtering operation. The resulting problem with the filtered Navier-Stokes equations is closed by modelling the residual stress

tensor. LES computational cost is certainly higher than RANS, but LES are more accurate for flow fields with significant large scale motions.

Direct Numerical Simulations (DNS).
This approach consists in solving the Navier-Stokes equations resolving all the scales (length scales and time scales) of motion. No turbulence model is needed in DNS. Clearly this approach is the most accurate and the computed flow field is fully described. The main drawback is the high computational cost: it is proportional to Re^3 so the use of DNS is currently limited to simple geometries at low Reynolds numbers.

In our simulations we will reproduce complex situations like the vortex shedding of working wind turbines and we are interested in the unsteady aerodynamic loads on the wind turbine structure as well as the evolution of the wake in the downstream flow. These phenomena take place in the large, energy-containing scales of motion, therefore large eddy simulations are the most suitable for this kind of problems: large scales are solved without models and so we can extract much more information about the flow field dynamics in comparison to RANS.

3.2.3 Loads estimation

The momentum equation 3.2 contains a source term to take into account the external forces. In our simulations the external forces are produced by the wind turbines, so we need to estimate them in an accurate way. Three approaches are possible:

Actuator Disc Model (ADM).
The wind turbine rotor is modelled as a disc and the forces as estimated from the local wind value on the entire rotor area. This model is very simple yet it produces good results, apart from the flow field close to the rotor region where the effects of the approximation are not negligible. Since the rotor is not moving, the actuator disc model can be used for steady simulations.

Actuator Line Model (ALM).
In this case each blade is modelled as a line with airfoil characteristics that rotates in the flow field, then each line is discretized into a finite number (typically 25-40) of elements in order to calculate the span-wise distribution of the loads. A sketch of the ALM is shown in Fig. 3.5. Each blade section (delimited by the yellow lines) has constant airfoil, chord and twist. This data is then used in a lookup table along with the local wind on the section to compute the aerodynamic forces in each actuator point (red dots). To avoid numerical oscillations the forces in the nodes cannot be be applied directly as a body-force to the momentum equations but they have to be projected onto the surrounding cells of the mesh. Compared to ADM, ALM is more representative of the real turbine dynamics as it can take into account the blade's airfoil properties and the pitch angle. It can produce blade tip vortices hence producing a better representation of the wake in the near-rotor region.

Fully resolved rotor method.
In this method the wind turbine is modelled directly into the CFD mesh. This implies that the numerical mesh must be very fine in the rotor region. This method is the most accurate but it is also the most computationally expensive [20].

In our simulations we will use the ALM because it can reproduce the rotor dynamics. By using the ALM, we can also retrieve the loads that act on each blade with enough accuracy to perform an analysis of the loads on the wind turbine structure. Furthermore it will not

add a relevant computational cost to the CFD simulation because it does not require a very fine mesh in the rotor area. If we were interested only in the far wake dynamics even the ADM would be a suitable choice.

3.2.4 Computational cost

Performing LES simulations of large wind farms with multiple wind turbine is a very demanding task in terms of computational cost and time. Such kind of simulation can not be run on personal computers with limited computational power. A high performance computer or a computer cluster is required in order to be able to run multi-core parallel simulations. Building a local HPC setup is expensive and requires continuous maintenance and upgrades, instead resorting to an existing remote cluster is a less expensive solution in the short term.

Figure 3.5: Actuator Line Model of a wind turbine rotor.

3.3 Case study

As noted by Wang, Bottasso and Campagnolo in [4] a side effect of the cyclical variation of pitch angle was the enhancement of the wake mixing, leading to a faster wake recovery and a higher downstream velocity. This particular phenomena was further investigated by Munters and Meyers in [5] where the vortex shedding from upstream turbines was modelled through a sinusoidal variation of the rotor thrust. The study showed that an increase in the overall energy production of the wind farm was possible, with a maximum increase of 5%.

We decided to exploit the wake enhancement mechanism by imposing a Periodic Collective Motion (further in the thesis we will refer to it as PCM) to the wind turbine blades, in particular by using the controller to superimpose a sinusoidal variation to the pitch angle of the blades. The blade pitch angle (equal for the three blades) will hence be given by the following expression:

$$\theta = \theta_{trim} + \theta_{PCM} \cos(\omega_{PCM} t) \tag{3.5}$$

$$\text{where} \quad \begin{cases} \theta_{trim} \text{ is the demanded pitch angle from the baseline controller} \\ \theta_{PCM} \text{ is the amplitude of the Periodic Collective Motion} \\ \omega_{PCM} \text{ is the angular frequency of the Periodic Collective Motion} \end{cases}$$

Munters and Meyers in [5] analysed the effects of the variation of operational parameters, identifying an optimal configuration that produced the best results in terms of overall power production. We decided to pick their optimal parameters as the starting point for our case study. The results reported in [5] express the overall power production increase as a function of the Strouhal number instead of the angular frequency ω_{PCM}.
The Strouhal number (St) is defined as:

$$St = \frac{f V_{wind}}{D} \qquad \text{with} \quad \begin{cases} D = 100\, m^1 \\ V_{wind} = 8\, m/s^1 \end{cases} \tag{3.6}$$

So the optimal value of $St = 0.25$ identified in [5] corresponds to a wake meandering frequency f equal to $0.02\, Hz$. Now we can easily compute the reference angular frequency ω_{PCM} that we have to impose:

$$\omega_{PCM} = 2\pi f = 0.13\, rad/s$$

It now must be pointed out that in [5] an actuator disc model is used. This implies that there is not a parameter analogous to the pitch angle. Therefore as reference pitch angle we supposed the value:

$$\theta_{PCM} = 2^\circ$$

We observed that this value produces a variation in the thrust coefficient C'_T with an amplitude that falls in the optimal range identified in [5]. Secondly we wanted to impose a limited motion the blade, in order to keep the induced loads and the deviation from the optimal pitch angle low.

[1] Values from the setup used by Munters and Meyers in [5]

Chapter 4

Tools

The tools chosen for performing the CFD simulations are SOWFA and FAST, two open-source software codes developed by the National Renewable Energy Laboratory (NREL) for wind farm applications.

SOWFA (Simulator fOr Wind Farm Applications) is a high-fidelity CFD simulation tool based on the OpenFOAM CFD software. It includes solvers and boundary conditions specific for wind energy applications, as well as wind turbine models.

FAST is an engineering tool for simulating the dynamic response of a multitude of horizontal axis wind turbines.

One of the main advantages of these two software is that they can be coupled to perform aerodynamic and structural analysis within the same simulation. This allows us to perform a wider range of simulations without the problem of interfacing different tools that were not designed to communicate with each other.

We also decided to use these software for their flexibility: they are both open-source so any missing or desired feature can be added easily.

Another reason that led us to the choice of SOWFA and FAST is the fact that they are currently used also by other members of the CL-Windcon consortium. Using the same software makes the collaboration easier and more profitable.

The simulations presented in this thesis were performed on Cineca's MARCONI high-performance computer.

4.1 Marconi HPC Cluster

SOWFA and FAST were installed on the MARCONI high performance computer run by SCAI (SuperComputing Applications and Innovation) department of CINECA consortium. The computer is based on Lenovo NeXtScale architecture using Intel Xeon and Intel Xeon Phi processors and running CentOS 7.2 as operating system.

Marconi is a top tier supercomputer, ranked 18^{th} in the June 2018 TOP500 chart, a list of the most powerful commercially available computer systems.

Marconi has an overall peak performance of about 20 Pflop/s, but an incoming upgrade should bring its peak performance to 50/60 Pflop/s.

We proposed this thesis to ISCRA (Italian SuperComputing Resource Allocation) for the C class call for projects. Once the project was evaluated positively we received 34125 processor hours on the Intel Xeon Phi partition (Marconi-A2).

4.2 SOWFA

SOWFA solves the three-dimensional Navier–Stokes equations 3.1 and 3.2 under the incompressibility assumption. Energy equation 3.3 is replaced by the potential temperature transport equation. The momentum equation takes into account the planetary rotation (Coriolis force), thermal buoyancy and the forces from the wind turbine.
The resulting incompressible Navier-Stokes equations are:

$$\frac{\partial U_i}{\partial x_i} = 0 \tag{4.1}$$

$$\underbrace{\frac{\partial U_i}{\partial t} + U_j \frac{\partial U_i}{\partial x_j}}_{\mathbf{I}} = \underbrace{-\frac{1}{\rho_0}\frac{\partial P}{\partial x_i}}_{\mathbf{II}} + \underbrace{\frac{\partial \tau_{ij}^D}{\partial x_j}}_{\mathbf{III}} \underbrace{-\frac{1}{\rho_0}\frac{\partial \Psi}{\partial x_i}}_{\mathbf{IV}} \underbrace{-2\varepsilon_{i3k}\Omega_3 U_j}_{\mathbf{V}} + \underbrace{\frac{1}{\rho_0} f_i}_{\mathbf{VI}} \tag{4.2}$$

Where 4.1 is the continuity equation and the terms in the momentum equation 4.2 are:

I is the material derivative of the fluid velocity U.

II is the pressure gradient contribution.

III is the deviatoric part of the viscous stresses tensor $\left(\tau_{ij}^D = \frac{\mu}{\rho_0}\frac{\partial^2 U_j}{\partial x_i \partial x_j}\right)$.

IV is the contribution due to buoyancy, with $\Psi = \rho_b g z$.

V is the contribution due to the Coriolis force, in particular Ω_3 is the rotation rate vector computed for the latitude of the chosen location.

VI is the contribution of the force from the wind turbine, that can be computed using an Actuator Disc Model (ADM) or an Actuator Line Model (ALM).

The potential temperature transport equation is:

$$\underbrace{\frac{\partial \theta}{\partial t} + U_j \frac{\partial \theta}{\partial x_j}}_{\mathbf{VII}} = \underbrace{-\frac{\partial q_j}{\partial x_j}}_{\mathbf{VIII}} \tag{4.3}$$

Where:

VII is the material derivative of the potential temperature θ. It is used instead of the absolute temperature because it simplifies the study of atmospheric stability.

VIII is the contribution from temperature fluxes.

When performing Large Eddy Simulations (LES), the above equations are filtered in order to remove the smallest scales of turbulence that will not be solved. The solver used for producing the results presented in this thesis is the so called $pisoFoamTurbine.ALMAdvancedFASTv8$, a LES solver that does not take into account Coriolis and buoyancy forces, so in the filtered equation the terms **IV** and **V** from equation 4.2 will not be present.

The filtered momentum and potential temperature equations for the $pisoFoamTurbine.ALMAdvancedFASTv8$ solver are:

$$\frac{\partial \bar{U}_i}{\partial t} + \frac{\partial}{\partial x_j}\left(\bar{U}_i \bar{U}_j\right) = -\frac{1}{\rho_0}\frac{\partial \bar{P}}{\partial x_i} + \frac{\partial \tau_{ij}^D}{\partial x_j} + \frac{\partial \tau_{ij}^R}{\partial x_j} + \frac{1}{\rho_0} f_i \tag{4.4}$$

$$\frac{\partial \bar{\theta}}{\partial t} + \frac{\partial}{\partial x_j}\left(\bar{U}_j \bar{\theta}\right) = -\frac{\partial q_j^R}{\partial x_j} \tag{4.5}$$

where the overline indicates the filtered quantites

As seen in chapter 3.2, the filtering operation generates a closure problem: models for the residual stresses τ_{ij}^R and the flux q_j^R are required.

Within the SOWFA LES solvers, the stresses and the temperature fluxes are modelled under the gradient-diffusion (or linear eddy viscosity) hypothesis using the Smagorinsky model for eddy viscosity:

$$\tau_{ij}^D = -2\nu_r\left(\frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i}\right) = -2\nu_r S_{ij} \tag{4.6}$$

$$q_j = -\kappa_r \frac{\partial \theta}{\partial x_j} \tag{4.7}$$

$$\text{with} \quad \kappa_r = \frac{\nu_r}{Pr_t}, \text{ where } Pr_t \text{ is the turbulent Prandtl number}$$

Both the above expressions require a definition for the residual eddy viscosity ν_r, that is given by the Smagorinsky model:

$$\nu_r = (C_S\Delta)^2\left[2\left(\frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i}\right)\left(\frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i}\right)\right]^{1/2} \tag{4.8}$$

By replacing 4.6 and 4.7 into respectively 4.2 and 4.3, we obtain a set of closed equations that can be solved numerically.

For the Smagorinsky Constant C_S values between 0.13 and 0.17 are used.

SOWFA solvers contain surface stresses models that are recommended over no-slip conditions for a better representation of Earth's surface roughness in atmospheric boundary layer simulations. The shear stresses at the surface and the temperature flux are modelled using the model proposed by Schumann. The model and its implementation into OpenFOAM are detailed in [8].

Depending on the solver, the numerical scheme used can be both the PISO (Pressure Implicit Splitting Operation) or the SIMPLE (Semi Implicit Method for Pressure Linked Equations) algorithms from OpenFOAM. The solver $pisoFoamTurbine.ALMAdvancedFASTv8$ clearly uses the former.

The PISO algorithm solves the Navier Stokes in the pressure-velocity formulation, following this procedure:

1. Initial U and P are chosen
2. Time step starts
3. Boundary conditions are updated
4. The linear system $\mathbf{M} \cdot U = -\nabla p$, that comes from the discretized 4.4, is solved
5. New U is obtained, so the mass flow over cell faces can be computed
6. The pressure equation is solved, giving matrix $\mathbf{A}$ and vector H
7. Pressure correction is computed from $\nabla \cdot \left(\mathbf{A}^{-1} \cdot \nabla p\right) = \nabla \cdot \left(\mathbf{A}^{-1} \cdot H\right)$.

8. Pressure correction is applied, giving new P
9. Mass flow over cell faces is corrected
10. Momentum correction $\left(U = \mathbf{A}^{-1} \cdot H - \mathbf{A}^{-1} \cdot \nabla p\right)$ is applied, giving new corrected U
11. Boundary conditions are updated
12. Steps 5 - 11 are repeated for the prescribed number of times
13. Time step ends

Projecting the forces calculated by the ALM into SOWFA's mesh is not a straightforward process and numerical problems may arise. SOWFA's approach to the problem is the use of smearing functions, in particular it implements a so called Gaussian projector:

$$f_i^T(r) = \frac{F_i^A}{\epsilon^3 \pi^{3/2}} \exp\Big[- \Big(\frac{r}{\epsilon}\Big)^2 \Big] \tag{4.9}$$

$$\text{where} \quad \begin{cases} F_i^A \text{ is force computed in the ALM node} \\ r \text{ is distance between the CFD cell center and the ALM node} \\ \epsilon \text{ is the smearing factor} \end{cases}$$

The tuning of the value of ϵ is crucial in order to avoid numerical oscillations.

4.3 FAST

Thanks to its modular interface, FAST can use different engineering models to retrieve many relevant properties of wind turbines. A scheme of FAST architecture is shown in Fig. 4.1. The core modules are the structural dynamics module, the control and electrical system module and the aerodynamic module.

The structural module uses different models for the elements of the wind turbine, the most relevant is the Geometrically Exact Beam Theory (GEBT) used for the blades, and performs a numerical solution through Finite Element methods.

The control module is very versatile and allows many different approaches: from simple Fortran control routines to an interface with MATLAB's Simulink.

In the case of SOWFA and FAST coupling, the aerodynamic module does not contain any model for the wind but it works as an interface between FAST and the wind velocity field computed in the SOWFA CFD simulation.

FAST can make use of other modules for very specific tasks, like simulating the effect of icing on the blades or computing the hydrodynamics of the waves in off-shore wind farms, which however will not be used in this work.

FAST uses nine different right-handed coordinate system for the input and output parameters:

Inertial Frame. The x axis points downwind and the z axis vertically opposite to gravity

Tower-Base. It is fixed in the support platform, the z axis points up from the center of the tower

Tower-Top/Base-Plate. It is fixed to the top of the tower and it does not rotate with the nacelle

Nacelle/Yaw. It is fixed to the top of the tower but it also rotates with the nacelle

Shaft. It is fixed at the base of the shaft with the x axis pointing along the shaft.It does not rotate with the rotor, but it translates and rotates with the tower. It yaws with the nacelle and furls with the rotor

Azimuth. It is located at the origin of the shaft coordinate system, but it rotates with the rotor

Hub. It is fixed at the intersection of the rotor axis and the plane of rotation, with the x axis pointing along the hub centerline. It rotates with the rotor and it can also teeter in two-bladed models.

Coned. It has the same origin of the hub coordinate system bu the z axis points along the blade pitch axis. There is one coned coordinate system for each blade.

Blade. Very similar to the coned coordinate systems with the difference that they pitch with the blades. Their origin is in the blade root.

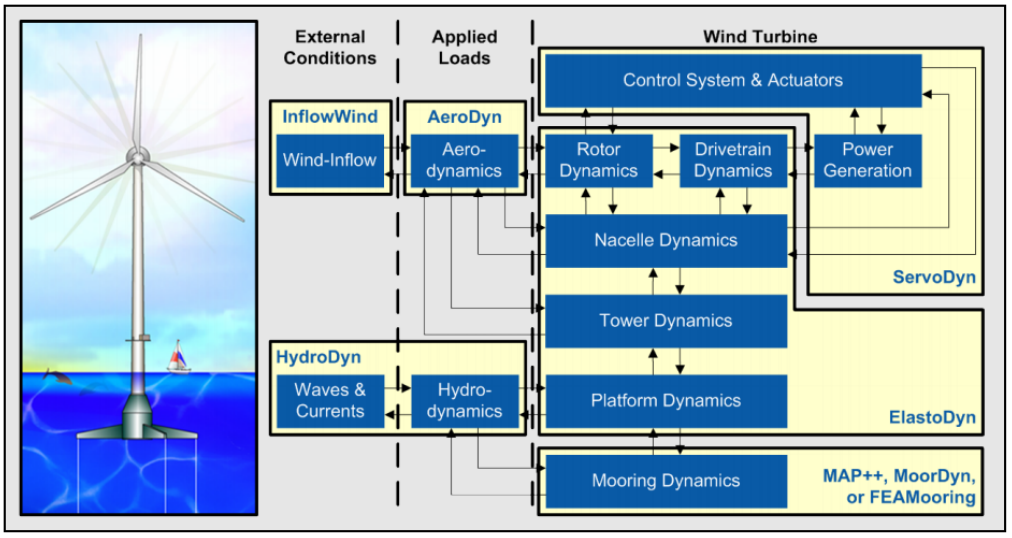


Figure 4.1: FAST v8 block scheme for floating off-shore wind turbines.

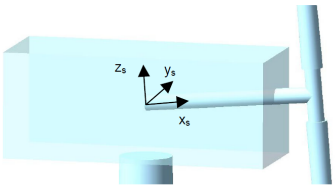


Figure 4.2: Shaft coordinate system. Picture from [7].

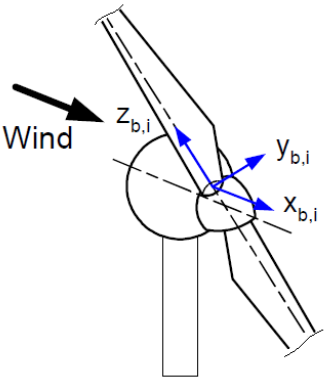


Figure 4.3: Blade coordinate system. Picture from [7].

4.4 Setup

Prior to the installation of SOWFA and FAST it is required to install OpenFOAM. In particular we installed OpenFOAM version 2.4.x, as it was reported as the latest release surely working with SOWFA, and all the required dependencies. Some of the dependencies were already available on MARCONI through the "modules" environment while others had to be compiled from the source code.

The FAST version with support for OpenFOAM and SOWFA coupling was downloaded from the specific GitHub's branch, but few minor changes to the install script were necessary in order to match the correct version of OpenMPI and the correct compiler available on MARCONI. In particular FAST was compiled with the GCC (GNU Compiler Collection) compiler using LAPACK library instead of Intel MKL.

The latest version of SOWFA was downloaded from GitHub as well but it needed some deeper changes before it could be installed on MARCONI, since some paths were hardcoded and had to be fixed according to MARCONI's directories. Furthermore some symbolic links and environment variables were missing and had to be defined correctly. Once all the correction were applied SOWFA was installed and ready to run simulations. All the steps performed to install SOWFA and FAST are reported in appendix A for future reference.

4.5 Validation

We performed some tests to validate the installation of SOWFA and check if it was running without any bug.

Install Validation: First of all we checked all the output logs for errors. This was actually a recursive process since the first install attempts gave many errors and it took several days of work to sort out all the problems and solve them. The particular architecture of MARCONI contributed to the difficulty of installation since it has a custom partitioning and system variables are not accessible from the end users. Once we overcame all the problems, the compilation succeeded without errors and we were able to run the provided installation test and verify that all the controls gave a positive result.

Test Cases: Once we were sure that SOWFA, FAST and all the related components were installed properly, we ran two provided test cases in order to verify the computed results and the performance of the MARCONI environment.

- First we ran a simulation using the Atmospheric Boundary Layer (ABL) solver simulating a 3km×3km×1km domain. This simulation did not included a wind turbine, so it was a SOWFA only simulation without FAST coupling. The simulation outcomes were compared to the ones presented in the 2014 SOWFA overview document [8]. Since the numerical output of NREL's simulation are not available, only a qualitative comparison was possible. Visualizations of the flow field from our simulation and from the NREL one are reported in Fig. 4.4 and 4.5. They are slices parallel to the ground taken at 50 meters height, the ABL has mean velocity of 9 m/s directed at 45°.

  More detailed data about the computational cost of the simulation on the NREL's system were instead reported, so a more accurate comparison could be drawn. The simulations were run under the same numerical conditions: same numbers of cores for the parallel execution, same domain, same number of cells and same timestep. The results in terms of clock hours used by the solver are reported in Tab. 4.1. We can conclude that the CINECA computer has a higher performance, thanks to the recent Intel Xeon processors.

- The second simulation used the PISO solver for wind turbine modelled as an Actuator Line and it was coupled with FAST. The aim of this test was to check that the SOWFA and FAST environments were coupled correctly and to check that using a new controller that we compiled on MARCONI was not generating issues. Once the simulation was over, a comparison of the flow field (see Fig. 4.6 and 4.7) showed identical results, meaning that our controller was working in the same way of the original without any sort of problem.

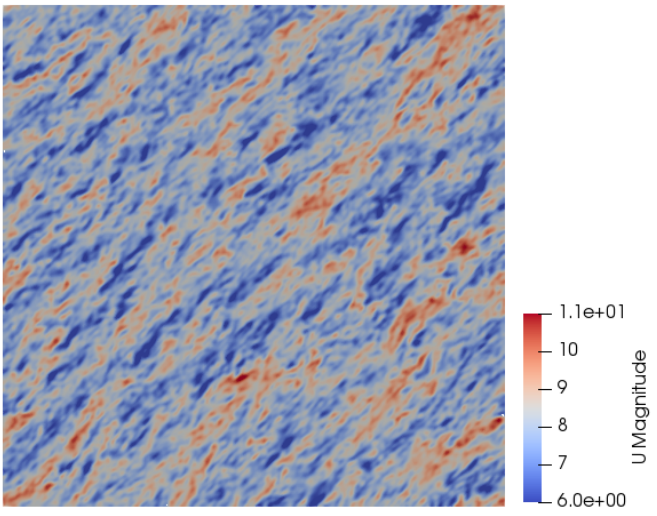


Figure 4.4: ABL flow field generated by our simulation.

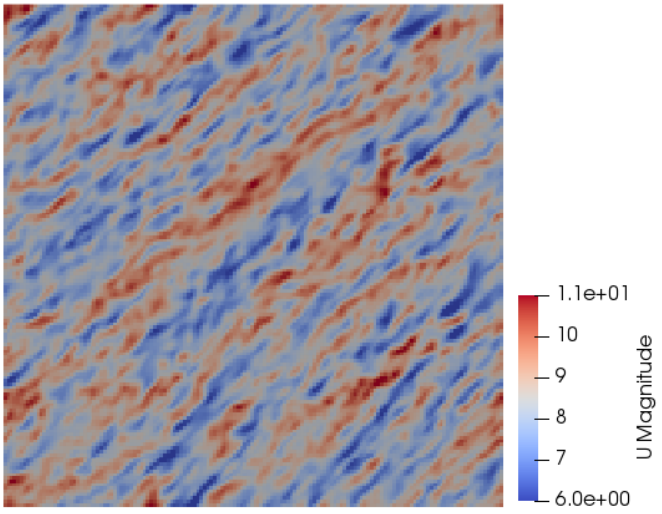


Figure 4.5: ABL flow field from NREL simulation [8].

	Simulation time [s]	**Clock time** [s]	**Computational performance** simulation time / clock time
NREL	1000	1620	0.62
MARCONI	1000	800	1.25

Table 4.1: Comparison between MARCONI and NREL HPC performance.

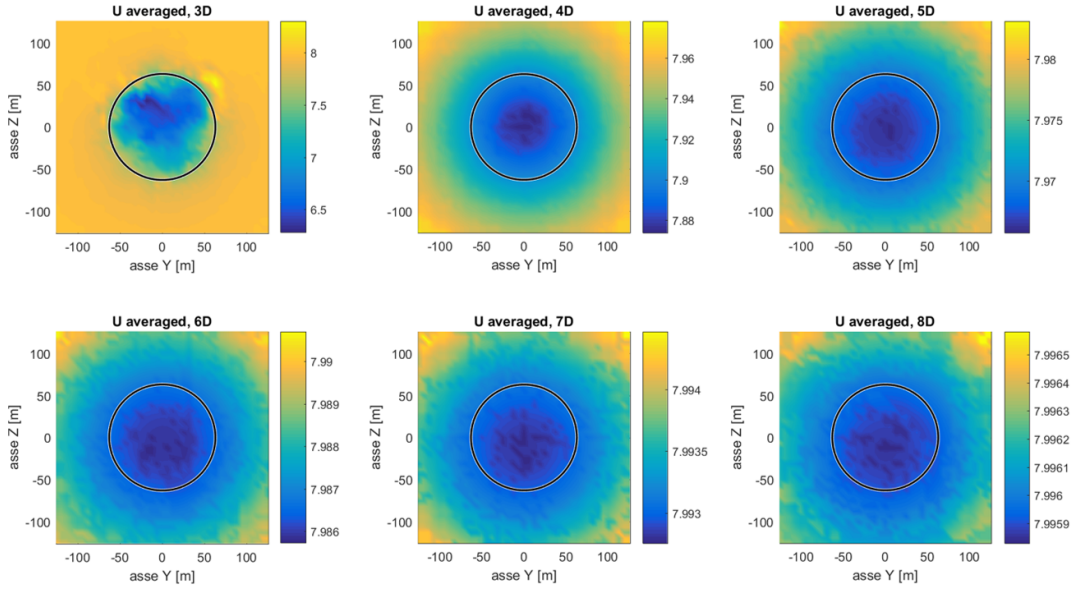


Figure 4.6: Simulation outputs with the controller compiled on MARCONI.

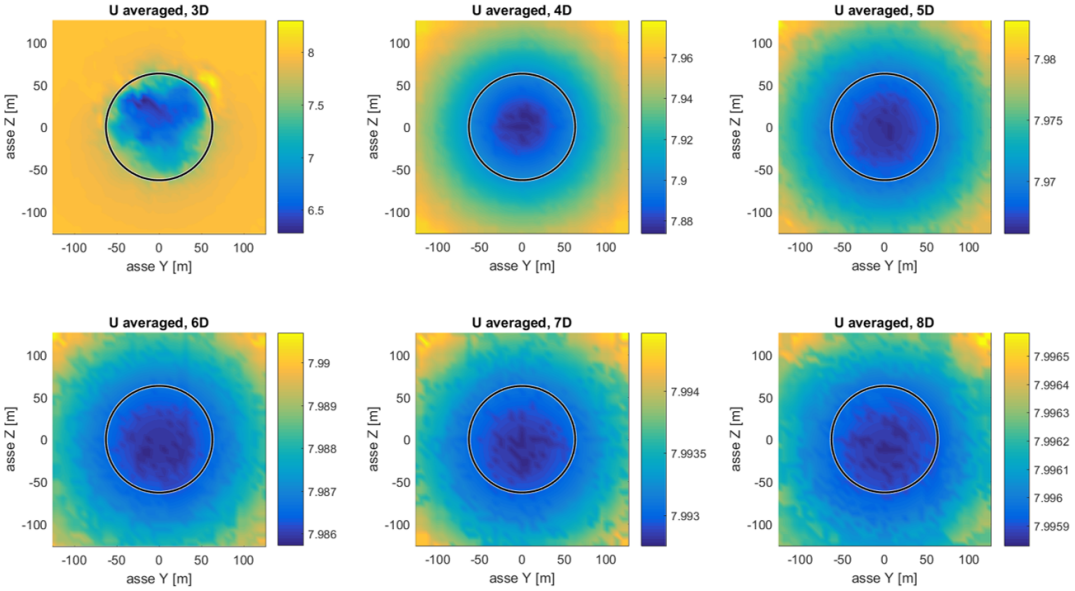


Figure 4.7: Simulation outputs with the provided controller.

Chapter 5

Models

In this thesis we simulated the behaviour of the flow around a wind turbine rotor, with particular interest for the downstream region. In this chapter we will detail the models used for the various elements of the numerical simulations.

5.1 Wind turbine properties

The wind turbine used in the simulations is based on the NREL 5MW off-shore turbine reference model [21]. It is a classical 3 bladed upwind turbine, with variable speed and blade pitch control system and a high speed multiple-stage gearbox.

The most relevant properties of the turbine are presented in Tab. 5.1.

Rated Power	[MW]	5
Rated wind speed	[m/s]	11.64
Rated rotor speed	[RPM]	12.1
Blade length	[m]	61.5
Rotor diameter	[m]	126
Shaft up-tilt	[deg]	5
Rotor pre-cone	[deg]	2.5
Cut-In wind speed	[m/s]	3
Cut-Out wind speed	[m/s]	25
Rotor mass	[kg]	110000

Table 5.1: NREL off-shore 5MW baseline wind turbine – Main properties.

The regulation trajectory used by the control system is shown in Fig. 5.1 and 5.2. It is possible to note that the regulation trajectory is divided into five different control regions:

Region 1 **(below $V_{cut\,in}$):** The wind speed is lower than the cut-in speed so the wind turbine is inactive.

Region $1\frac{1}{2}$ **(3 m/s - 8 m/s):** The wind turbine is starting up, the aerodynamic torque is controlled in order to guarantee that the rotation speed ramp is not too steep.

Region 2 **(8 m/s - 10.75 m/s):** The wind turbine is producing energy, but due to the fact that the wind speed is lower than the rated wind speed the production is less than 5 MW. The control system is designed to maintain the optimal value of tip-speed-ratio, in order to guarantee the maximum value of C_P, typically indicated as C_P^*.

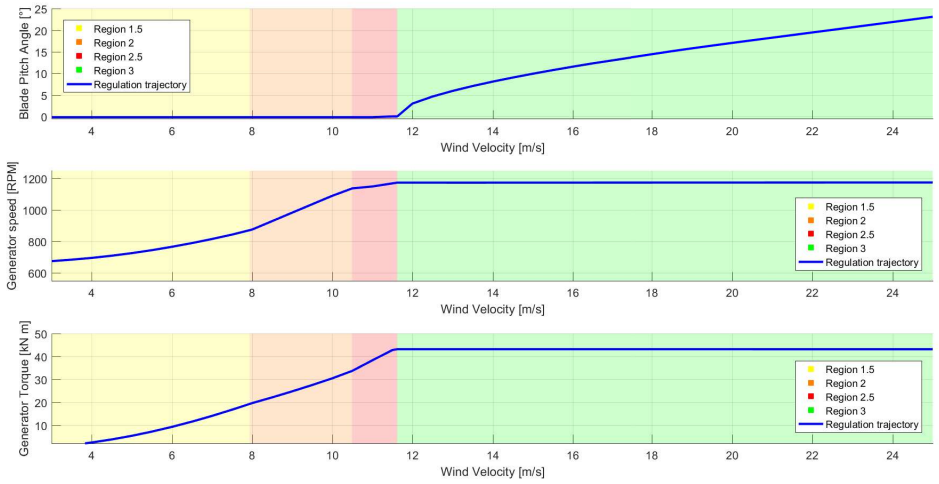


Figure 5.1: Controller regulation trajectory – Pitch angle, rotation speed, torque.

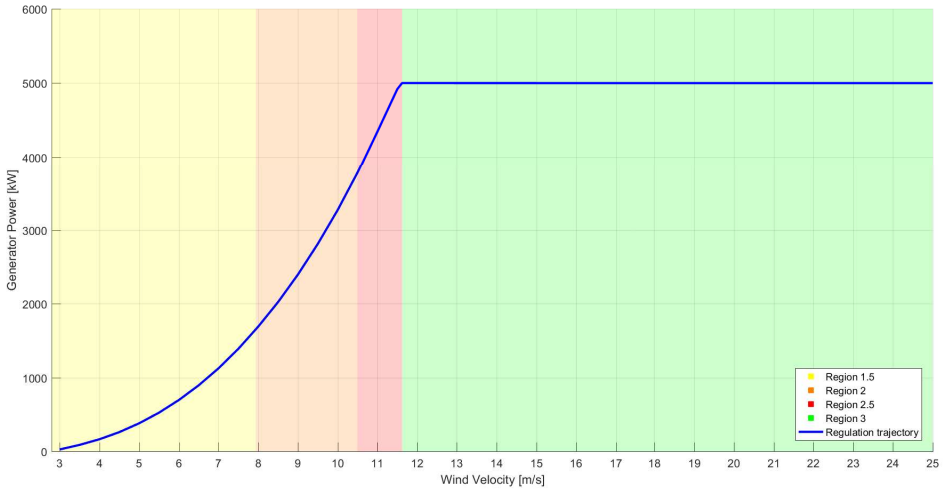


Figure 5.2: Controller regulation trajectory – Nominal power curve.

Region $2\frac{1}{2}$ **(10.75 m/s -** V_{rated}**):** Operating at the maximum value of C_P is not possible due to excessive noise emission, so the controller is designed to reduce the tip-speed-ratio to a sub-optimal value.

Region 3 **(**V_{rated} **-** $V_{cut\,out}$**):** The wind turbine is in full production thanks to wind speeds above the rated value, the controller maintains the generated power equal to the nominal value of 5 MW.

5.2 Domain properties

The domain of the simulation is a cube with sides equal to 20 times the rotor diameter, this results in a domain of dimensions 2520×2520×2520 meters. The origin of the reference

axes is fixed in the geometric centre and the coordinate system is right-handed. Each coordinate therefore spans from -1260 to +1260 meters.

The wind turbine rotor is placed in the origin of the coordinate system and the shaft is up-tilted of 5° with respect to the x axis. The wind turbine tower and nacelle are not present, so their effect on the flow will be neglected.

A schematic of the domain is shown in Fig. 5.3

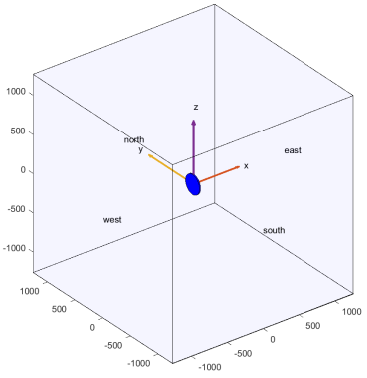


Figure 5.3: Domain layout.

5.3 Wind properties

After analysing the regulation trajectory of the wind turbine we decided to perform the simulation with the turbine operating below the rated wind speed. The main reason why we didn't choose to use a working point from region 3 is because for such values of wind speed the wind turbines will likely be producing the nominal power, so a better wake recovery would not lead to a higher power production, as the control system does not allow it. On the other hand, in region $1\frac{1}{2}$ and 2 the power production grows proportionally to the third power of the wind speed, so even a small increase in the downstream velocity could lead to a sensitive increase in power production.

We also wanted to avoid the transition region $2\frac{1}{2}$, since it is not always present in the regulations trajectories and it is not a typical working condition for wind turbine.

All these reasoning led us to the choice of 6.5 m/s as reference wind speed. The wind direction is parallel to the x axis, from west boundary toward the east boundary.

The wind is imposed in the simulation through boundary conditions, in particular on the west boundary it is imposed the exact value of 6.5 m/s, while on the east boundary the outflow is imposed with a zero gradient condition on the flow velocity. On the remaining four boundaries (north, south, upper and lower) a cyclic boundary condition on the velocity is imposed. This means that the wind shear due to the atmospheric boundary layer is neglected.

The Earth rotation is also neglected, so the contribution to the flow given by the Coriolis forces is null and the wind is not veering.

In this thesis we focused only on the effects of the PCM control technique: we considered only one turbine rotor and the domain does not reproduce an existent wind farm. So even for the flow field, we neglected as many external factors as possible, in order to observe only phenomena related to the PCM of the blades. More realistic simulation focusing on the

interaction between PCM, possibly implemented on multiple wind turbines, and atmospheric boundary layer or turbulence generated by environment elements are an interesting future development of this thesis.

The pressure field is imposed using the same typology of boundary conditions of the velocity field using zero as reference value.

The turbulence is initially set to zero in terms of both turbulent kinetic energy and turbulent viscosity.

5.4 Numerical parameters of the simulation

The numerical mesh was generated starting from a coarse uniform mesh with 80 cells per side of the domain. It was then refined in order to achieve a satisfactory density of cells in the wake region. We performed four sub-sequential levels of refinement using the $refineHexMesh$ utility from OpenFOAM. A schematic of the refinement regions is shown in Fig. 5.4, the level 1 refinement region has been refined only once while the level 4 refinement region has the finest mesh since it has been refined four times. In Fig. 5.5 and 5.6 are reported slices of the mesh in the yz plane, where it is possible to note the different size of the cells in different refinement regions.
The resulting mesh has 6.77×10^6 cells, of which 6.65×10^6 are hexahedral cells and the remaining are polyhedral.

The timestep used in the SOWFA simulation is fixed to 0.048 seconds, while the FAST simulation used a timestep of 0.016 seconds. This difference is due to the fact that FAST performs an interpolation over an interval of three timesteps, so the outputs will be saved every 0.048 seconds in perfect synchrony with the SOWFA simulation. The simulation time was set to 600 seconds, enough to observe a fully developed wake. We investigated a shorter simulation time of 180 seconds, in order to reduce the computational cost of each simulation, but we concluded that it was not suitable because the wake in the far downstream region was not fully developed. The Courant number limit is set to a maximum value of 0.75, but among all the simulation it never exceed values of 0.3/0.35.

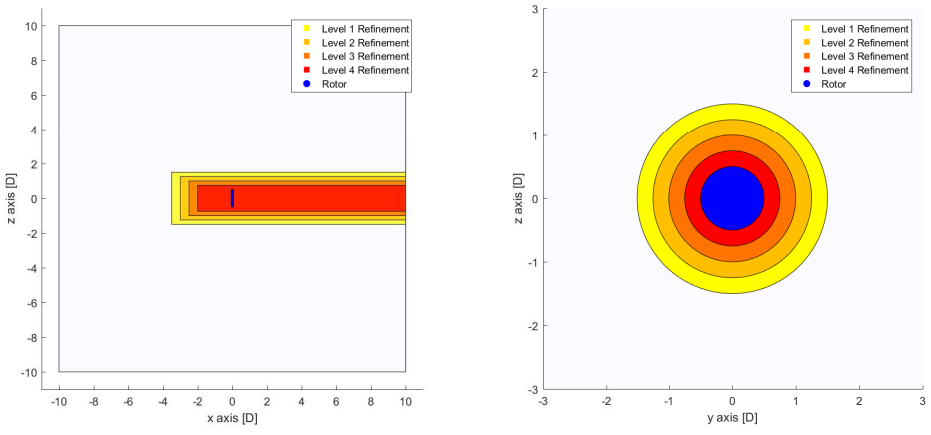


Figure 5.4: Refinement Regions in the x-z and y-z planes.

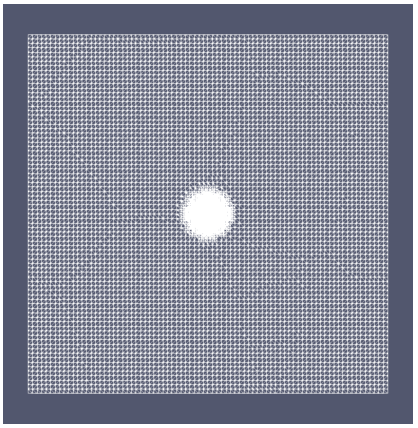


Figure 5.5: Numerical mesh – y-z plane view.

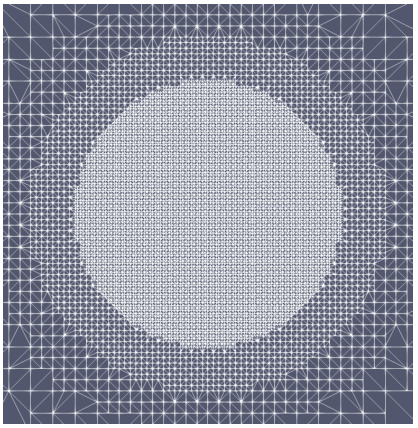


Figure 5.6: Numerical mesh – close-up view of the refined region.

5.5 Velocity Probes

In order to gather data in a more handy way, we set up an array of probes to collect data about the flow velocity. To do so we exploited the *Probes* class of OpenFOAM sampling functions, which identify the closest cell to the selected probe location and give back the value of the requested field.

We set up a grid of probes that spans a square of side equal to 2 diameters. Putting 41 probes in each direction, we obtained a resolution of 1 velocity measurement every 6.3 meters, both in x and y directions.
We placed the grids of probes along the x axis, downstream at distances of 3, 4, 5, 6, 7, 8 diameters.
A scheme of the probes setup is shown in Fig. 5.7. We decided to save the value of velocity read by the probes every 10 timesteps, which corresponds to a sampling frequency

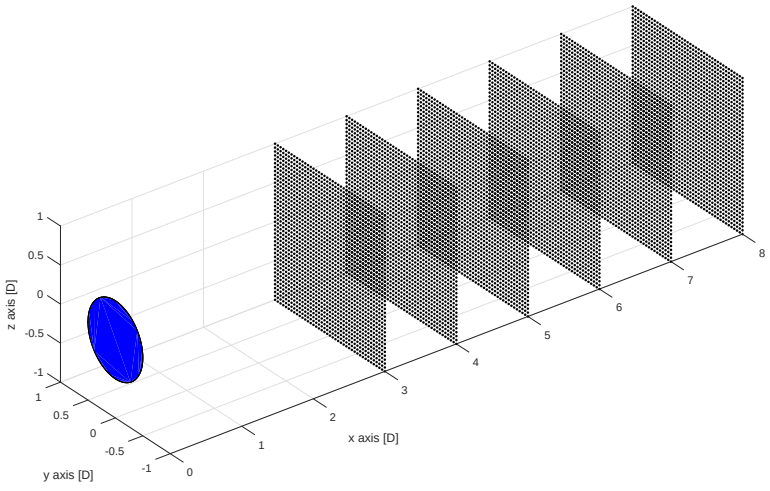


Figure 5.7: Sampling probes setup.

of approximately 2 Hz. The output file generated by the probes grid has a size of about 100 MB for the whole simulation, which is much more manageable than the 4 GB output generated by OpenFOAM at every time instant. The output file was then postprocessed using MATLAB.

Chapter 6

Simulations results

6.1 Effects of periodic collective motion on the wake

First of all we want to compare the wake downstream the turbine with the standard control strategy to the one when the PCM control is implemented, in order to evaluate whether the wake is actually recovering faster and eventually the magnitude of this phenomenon.

Figure 6.1 shows the behaviour of the wind turbine in terms of blade pitch angle, rotor speed and generated power for the two controllers. The transient caused by the startup of the wind turbine at the beginning of the simulation is not reported in the plots. From the pitch angle plot it is possible to verify that the controller actuates very smoothly the desired sinusoidal variation (equation 3.5). The sinusoidal behaviour is also reflected by the rotor speed and the generator power. As we will see, the PCM periodicity will be present in many measured quantities.

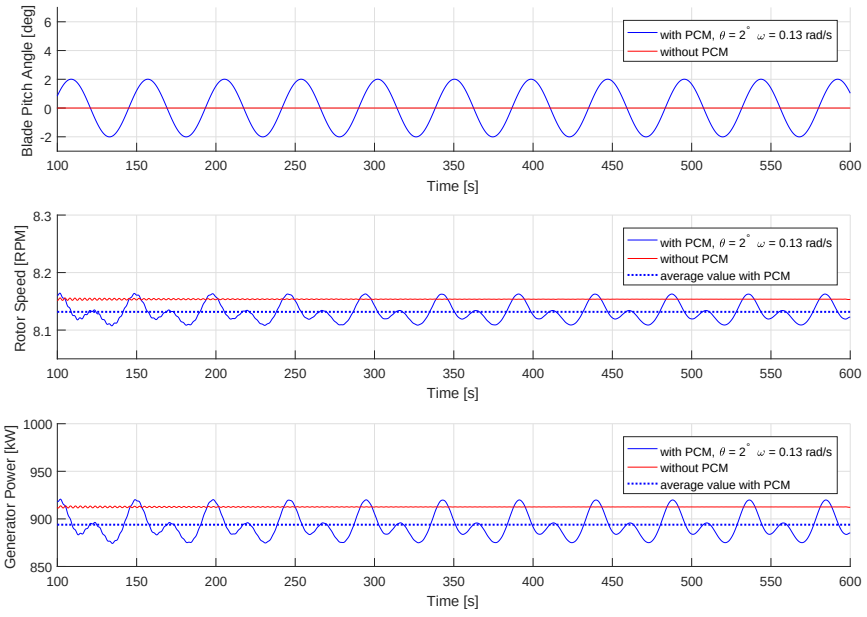


Figure 6.1: Effects of PCM on blade pitch angle, rotor speed and power production.

From the generated power plot, we can note that the wind turbine power production is reduced when the PCM is implemented. This drawback is common to all wind farm control techniques: since the standard controller in region 2 is designed to work in the optimal power production condition, every deviation from the original control logic will cause a local loss in the generated power. However the advantage of PCM is a reduced decrease in the power production of the upstream turbine with respect to other control strategies, leaving a higher margin for an overall increase. For example, in axial induction control techniques power production losses in the upwind turbine have been measured in the range of 15% - 20% [3]. In wake redirection techniques by yaw misalignment losses about 10% have been estimated [4].

Figures 6.2 and 6.3 show the instantaneous velocity of the wake downstream the rotor, at time equal to 600 seconds, with standard controller and with PCM control. The values of velocity are those sampled using the velocity probes as explained in chapter 5.5.

Due to the turbulence of the flow, the wake region is not well defined in the instantaneous plot of the velocity, so we decided to analyse the average velocity of the flow. Furthermore, the main objective of wake recovery techniques is to increase the wind farm power production which is primarily related to the flow mean velocity; in a long time span the effects of small velocity fluctuations related to turbulence have a negligible effect. We defined the average velocity U_{ave} as the time average of the streamwise velocity above an interval of 3 minutes, in particular from simulation time of 420 seconds to 600 seconds.

$$U_{ave}(y,z) = \frac{1}{\Delta T}\int_{T_{final}-\Delta T}^{T_{final}} U_{rotor}(t)\,dt \qquad \text{with} \quad \Delta T = 180\ [s] \tag{6.1}$$

The average velocity is shown in Fig. 6.4 and 6.5.

Analysing the average flow velocity, it is possible to note that the wake region is smoother when the PCM is implemented, this suggests that the objective of enhancing the wake mixing is met and the low velocity region generated by the blades tip is recovered faster. As a consequence the values of average velocity are higher when PCM control is implemented and it seems to produce the expected result of reducing the velocity deficit due to the wind turbine wake.

The plots presented so far are very effective for visualizing the flow field, but they're not very handy when it comes to quantifying the effect of the PCM.

In order to directly compare the flow velocity for both the fields, we decided to plot the velocity profile along the z axis. The velocity profile was obtained from the measurements of central column of probes (the one with coordinate y=0). The velocity profile was averaged on a time interval of 3 minutes, just as above. Figure 6.6 shows the velocity profile for all the measuring stations. Near the turbine (at a distance of 3 diameters) the wake of the rotor without PCM shows a very slow region close to the blade tip but it is faster in the core region, while the wake of the rotor implementing PCM is smoother and has an almost constant value of stream velocity from the centre to the edge. A magnification of this measurement station is reported in Fig. 6.7. Considering a distance of 6 diameters, which is a typical distance for the installation of two aligned turbines in a wind farm, we can appreciate the effect of the PCM on the wake: looking at Fig. 6.8, the velocity profile generated by the rotor with PCM is higher along all the wake region. This means that the downstream turbine faces a higher wind and so it will be able to generate a higher power.

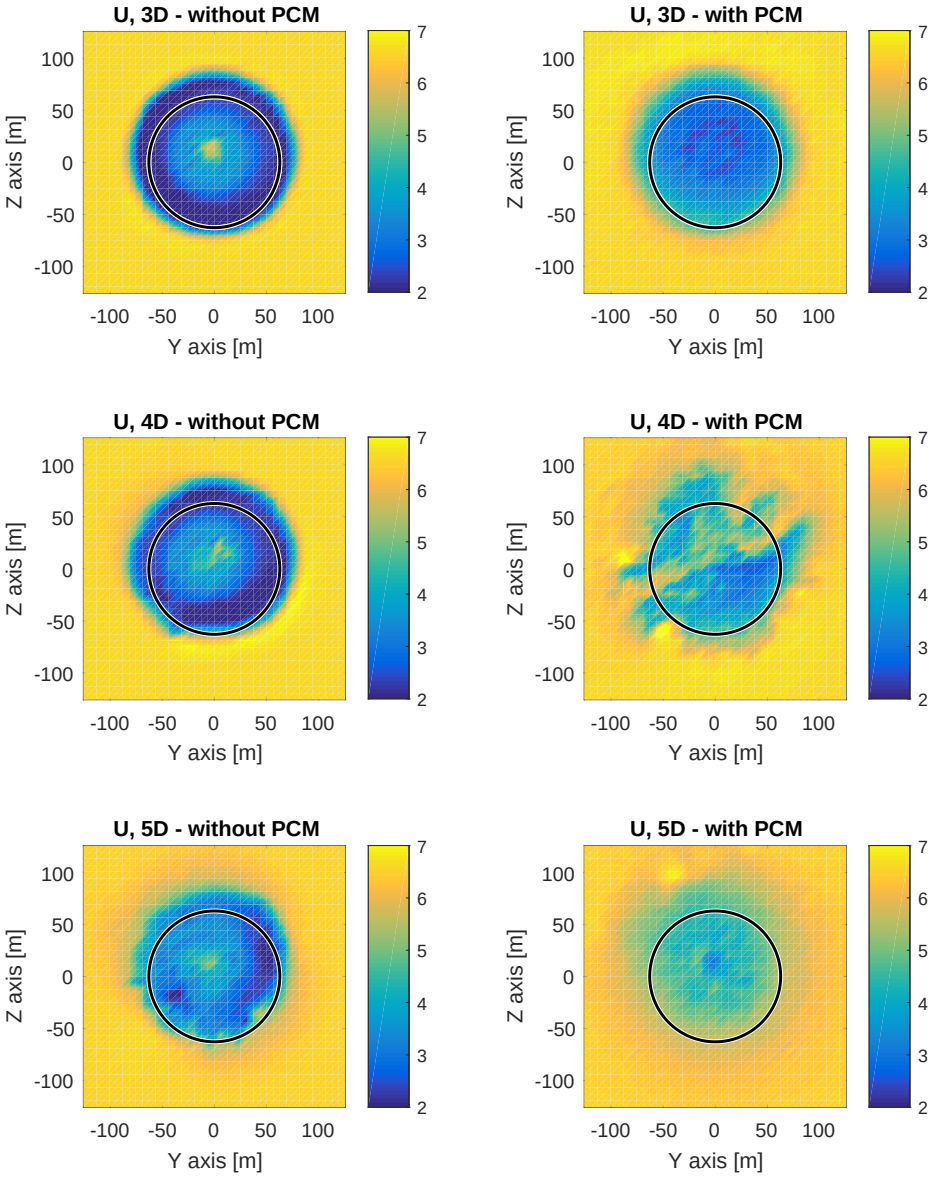


Figure 6.2: Instantaneous flow velocity at 3, 4 and 5 diameters – without PCM (left) and with PCM control (right).

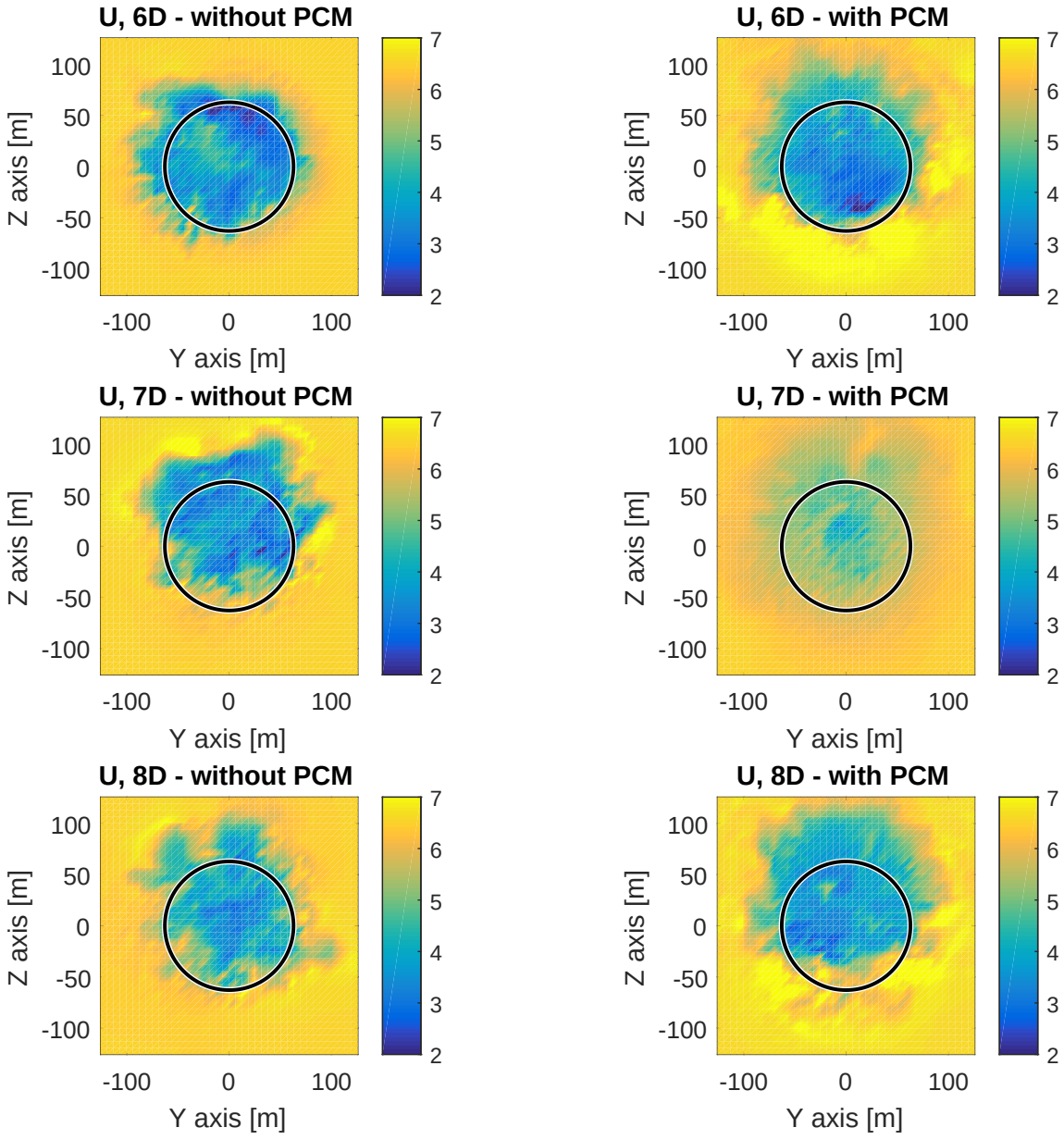


Figure 6.3: Instantaneous flow velocity at 6, 7 and 8 diameters – without PCM (left) and with PCM control (right).

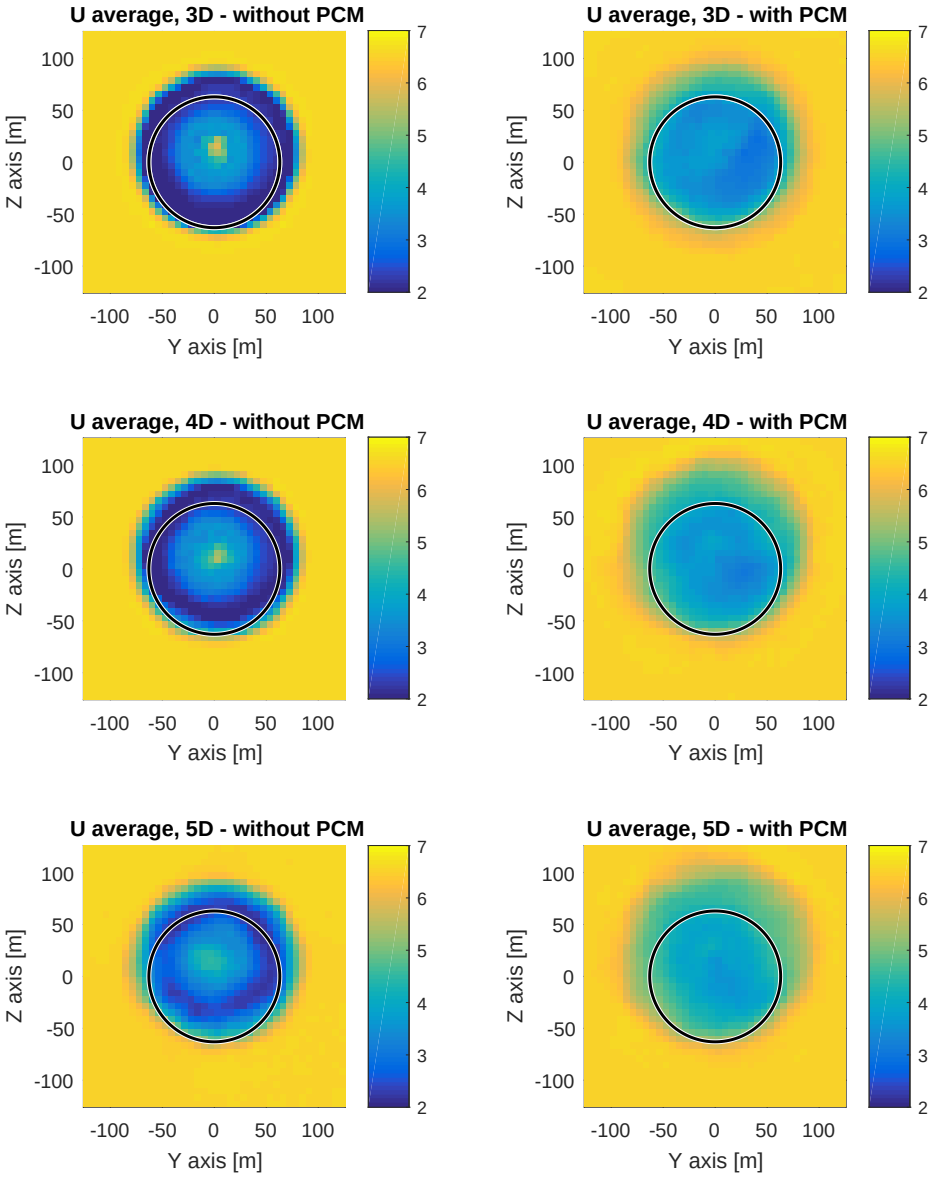


Figure 6.4: Average flow velocity U_{ave} at 3, 4 and 5 diameters – without PCM (left) and with PCM control (right).

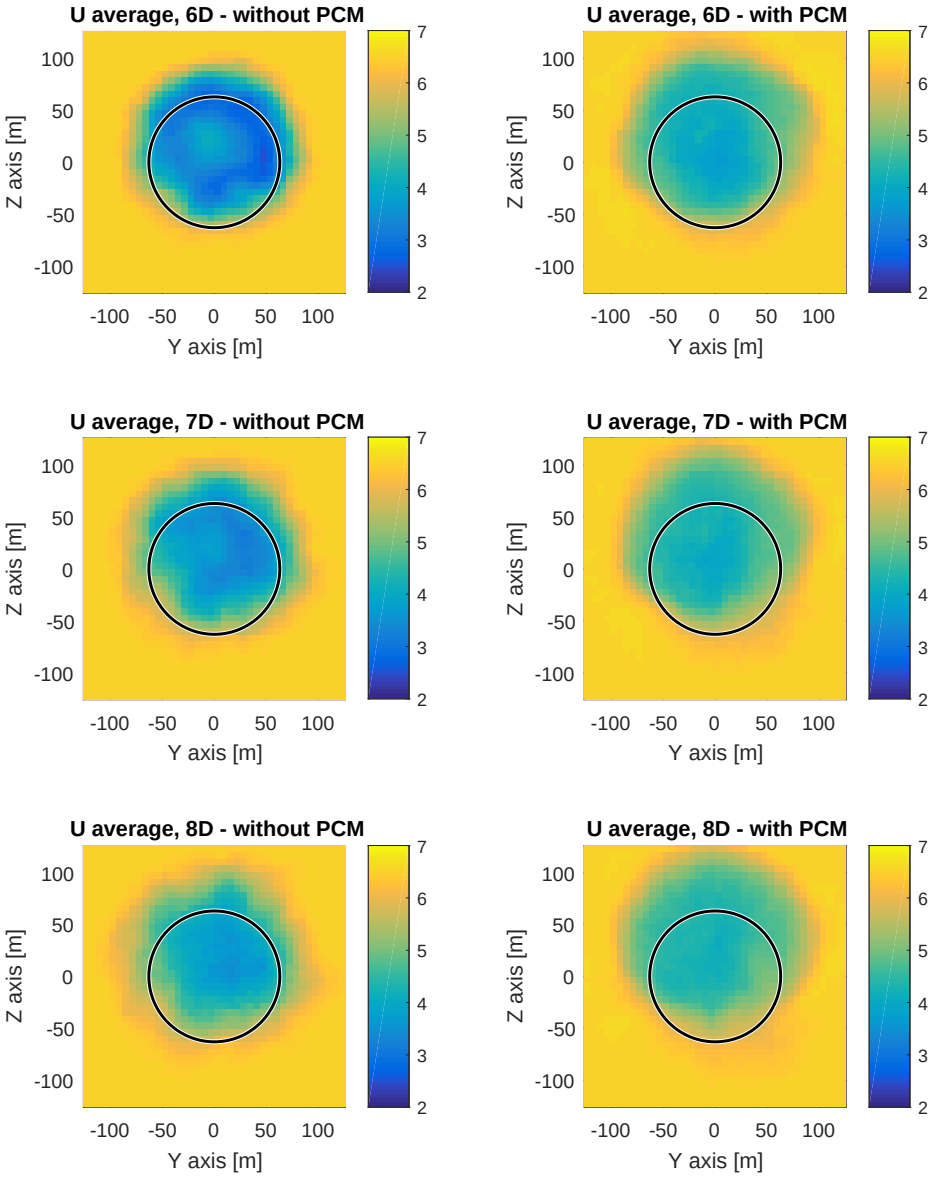


Figure 6.5: Average flow velocity U_{ave} at 6, 7 and 8 diameters – without PCM (left) and with PCM control (right).

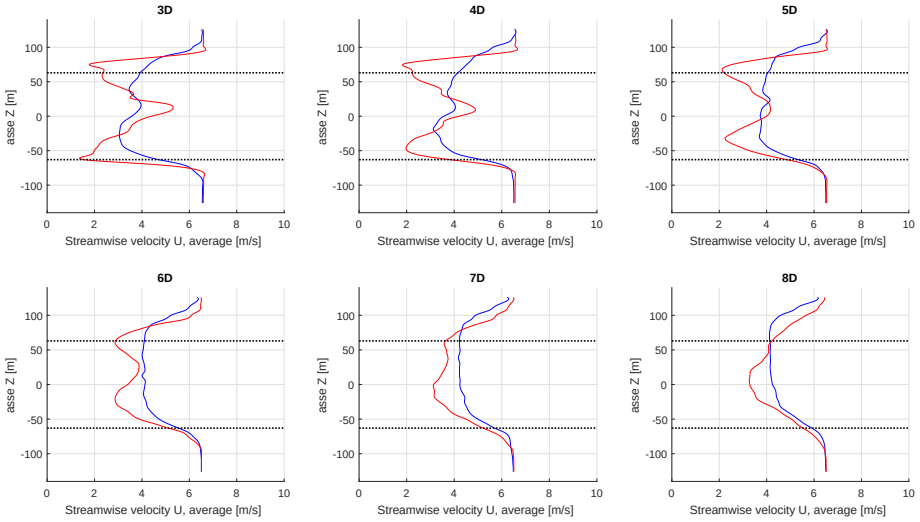


Figure 6.6: Velocity profile – red dashed line: without PCM, blue line: with PCM.

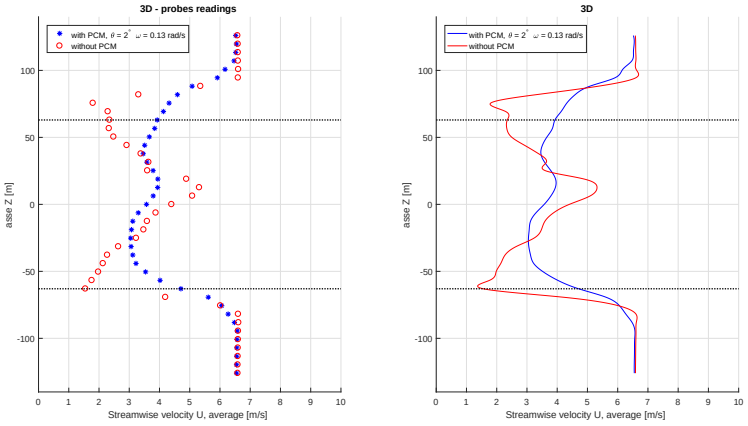


Figure 6.7: Velocity profile at 3 diameters – measured values and interpolated curve.

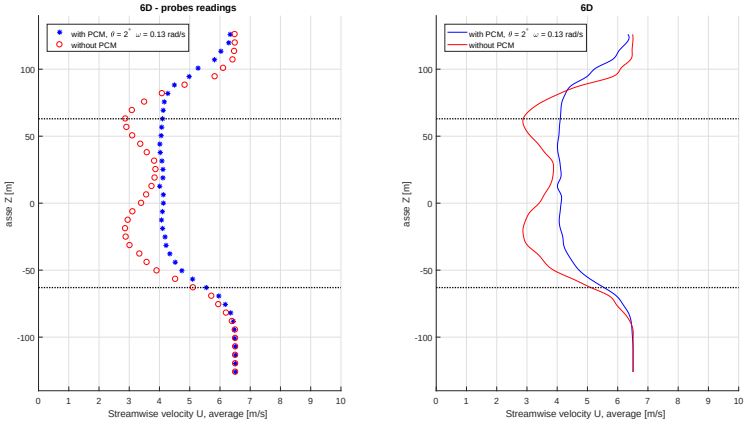


Figure 6.8: Velocity profile at 6 diameters – measured values and interpolated curve.

6.2 Parametric Analysis

In the previous section we showed that a periodic collective motion of the rotor blades is an effective approach to speed up the wake recovery and hence reduce the streamwise velocity deficit in the wake region. In this part of the thesis we will investigate the effects of the variation of frequency ω_{PCM} and amplitude θ_{PCM} on the wake behaviour, with the aim of finding the optimal combination of the two that guarantees the greatest wake recovery.

First of all, we investigated how the wake velocity varies with the PCM frequency ω_{PCM}. Figure 6.9 shows the velocity profile for five different values of frequency from 0.0975 to 0.2275 rad/s.

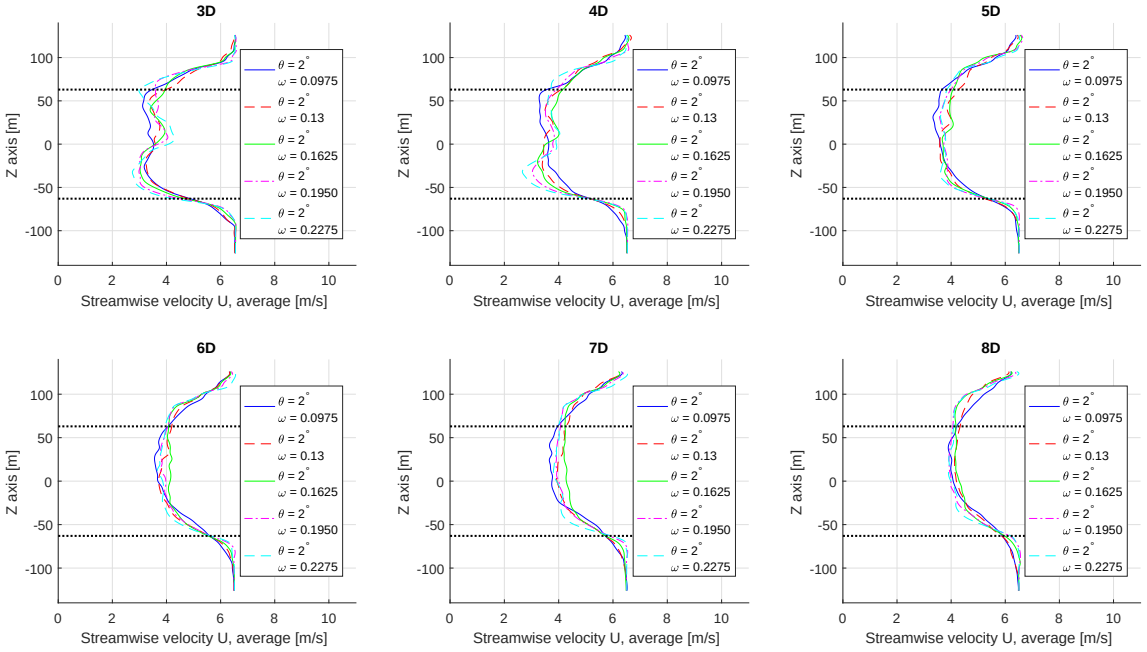


Figure 6.9: Velocity profile – values for different frequencies ω_{PCM}.

Analysing the plots, with particular attention to the stations at 5 and 6 diameters, we realized that the highest and lowest frequencies (0.0975 and 0.2275 rad/s) give the worst results in terms of streamwise velocity. On the other hand, the frequencies of 0.13 and 0.1625 rad/s produce the lowest velocity deficit in the wake region, and hence seem to be the optimal choice for ω_{PCM}. Taking a closer look (Fig. 6.10) to the velocity profile of those two frequencies, at a distance of 5 and 6 diameters, it appears that the optimal value is 0.1625 rad/s since it produces a slightly lower velocity deficit in the rotor region.

Looking at the velocity profile gives important information about the velocity of the wake that impacts on downstream turbines, also it gives us an idea about the shape and the evolution of the wake. For example, we can notice that the wake is moving upwards (toward positive z values) as it propagates downstream. This fact should be caused by the rotor being uptilted, that, as a reaction, impresses a force to the flow that deviates it upward.

Unfortunately the velocity profile does not express the overall wake velocity deficit and it is not a suitable measure for a direct comparison, for example it can happen that different profiles cross each other making it difficult to estimate which produces an overall better wake recovery. That's why we decided to define a new quantity to expresses the average wake velocity.

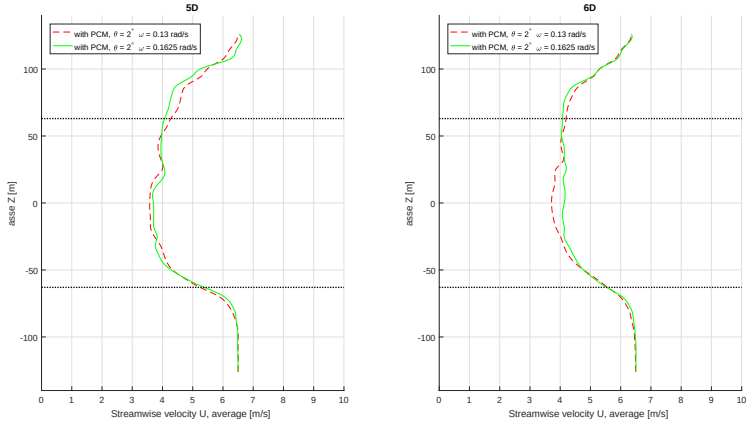


Figure 6.10: Velocity profile at 5 and 6 diameters – values for different frequencies ω_{PCM}.

First of all we took the average of the velocity in the rotor area:

$$U_{rotor}(t) = \frac{1}{A_{rotor}} \iint_{A_{rotor}} U(y,z,t)\, dy\, dz \qquad \text{with} \quad A_{rotor} = \pi\left(\frac{D}{2}\right)^2 \tag{6.2}$$

This operation produces the average value of the streamwise velocity for each time step, that is reported in Fig. 6.11 for all the different ω_{PCM} at a distance of 6 diameters.

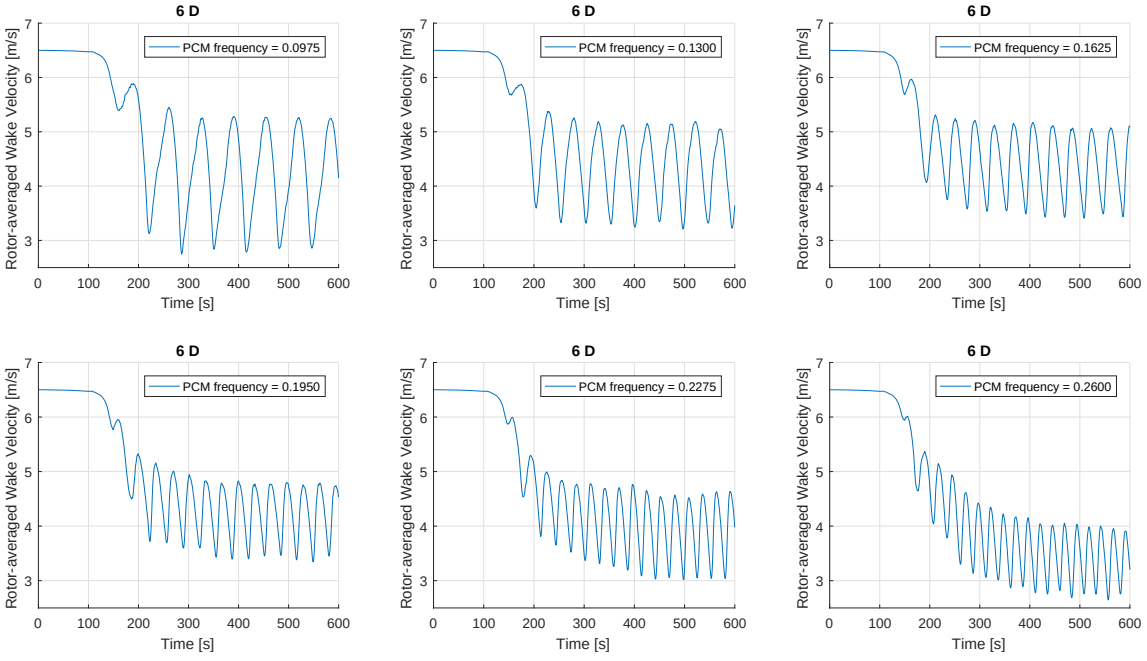


Figure 6.11: Rotor-averaged instantaneous velocity – values for different frequencies ω_{PCM}.

Then we took a time average of the rotor-averaged velocity, averaging above the last 3 minutes of the simulations as in the previous section:

$$\text{Average Wake Velocity} = \frac{1}{\Delta T} \int_{T_{final}-\Delta T}^{T_{final}} U_{rotor}(t)\, dt \qquad \text{with} \quad \Delta T = 180\ [s] \tag{6.3}$$

This gives us one single performance index for each simulation: the higher the average wake velocity, the lower is the streamwise velocity deficit which means a more effective wake recovery.

Using the average wake velocity as performance index, we can evaluate the effect of the variation of the PCM amplitude θ_{PCM}, in particular we investigated a higher angle of 2.5° and a lower angle of 1.5°.

Figure 6.12 reports the results of our parametric analysis for all the six measurement stations. Since we are mainly interested in the increase of wake velocity with respect to the case without PCM control, we decided to normalize the performance with the average wake velocity of the reference case without PCM.

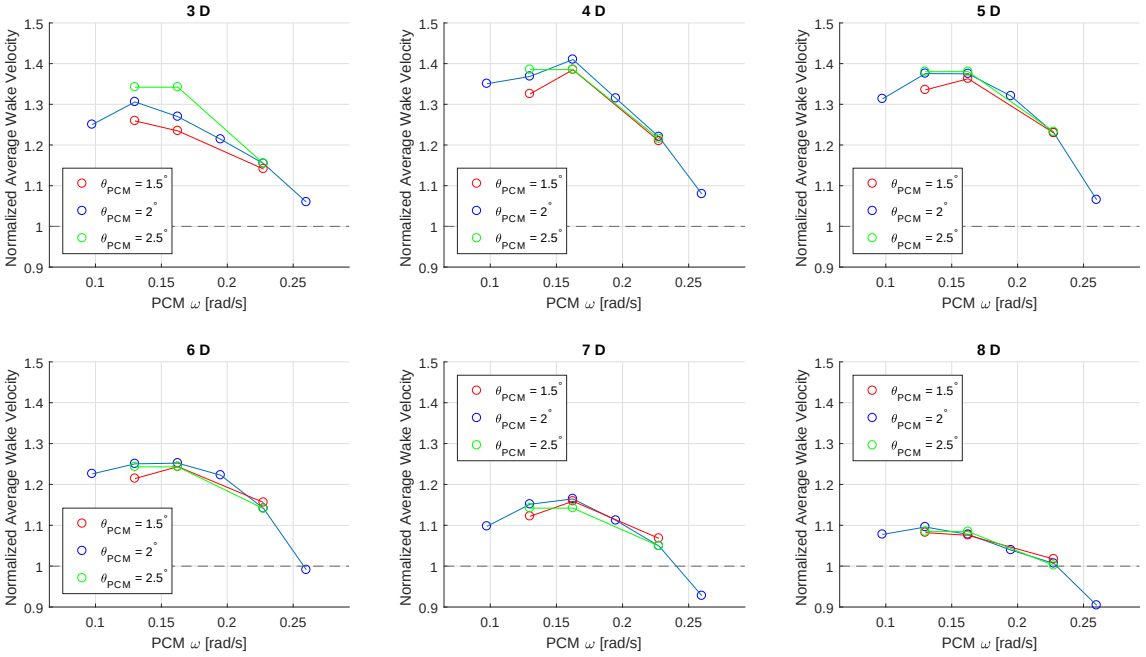


Figure 6.12: Normalized average wake velocity – values for different frequencies ω_{PCM} and amplitudes θ_{PCM}.

The distance between two adjacent wind turbines in a wind farm can range from a minimum of 3 diameters to over 8 diameters, depending on the location of plant and the particular environmental conditions. We will focus on the range of distances from 4 to 6 diameters because they are the most common in existent commercial wind farms. The optimal choice of PCM parameters for these distances is:

$$\begin{cases} \omega_{PCM} = 0.1625 \ rad/s \\ \theta_{PCM} = 2° \end{cases} \tag{6.4}$$

The optimal values are close to those predicted in [5]. It is possible to note that different combinations of ω_{PCM} and θ_{PCM} in the neighbourhood of the optimal values produce similar wake velocities. Also this behaviour is in concordance with Munters and Meyers results. This fact has positive implications: sub-optimal choices of PCM parameters can be taken into account as they will produce average wake velocities close to the optimal one.

6.3 Comparison between FAST-coupled SOWFA simulations and standalone FAST simulations

Before proceeding with more detailed results, it is important to underline the differences between FAST and SOWFA coupled simulations and aero-servo-elastic simulations using only FAST.

The main difference between a standalone FAST simulation and simulations where FAST is coupled with SOWFA is the wind flow. In coupled simulations the entire flow field is computed through a LES by SOWFA and passed to FAST. In standalone simulations a wind grid is generated using different models and the local flow velocity on the rotor is then computed using Blade Element Momentum Theory (BEMT). We performed simulations at the fixed trim point corresponding to the inflow wind speed of 6.5 m/s and compared the results produced by SOWFA and standalone FAST. As expected the two simulations produced different results. It should be noted that for both the simulations the results are very sensitive to the precise choice of models and parameters. In particular we ran the standalone FAST simulations with a steady wind grid and taking into account tip and hub losses in the BEMT induction model. A fundamental parameter in the SOWFA simulations is the smearing factor ϵ (see equation 4.9). Ideally this parameter must be tuned to obtain a perfect match between the simulation results and measurements on the real wind turbine. Since the NREL 5MW wind turbine is only a theoretical model there is no experimental data to validate the results. So we decided to use the values for ϵ suggested in literature [8]. In particular we picked a value of ϵ equal to 2.2 times the size δx of the mesh around the blades.

The results are reported in Fig. 6.13 and 6.14, where the red line refers to the SOWFA and FAST coupled simulation with reference value of ϵ and the blue line refers to the standalone FAST simulation. Comparing the results of the different simulations we can measure a discrepancy in the generated power in the range of 10-15 %. The results concerning the thrust force on the rotor show a smaller variation, about 5%, between the simulations. Even in terms of loads a difference in the order of 5% is measured.

Since the wake meandering is directly related to the thrust applied by the rotor, we can assume the same accuracy of about 5% also for the wake velocity. It is worth mentioning the fact that the suggested value of ϵ produces power coefficients very close to the Betz limit of $\frac{16}{27}$. This suggests that for our particular case the ϵ value might be too high. Taking the standalone FAST simulation as a reference, we performed SOWFA coupled simulations with different values of smearing factor and we noted that, for our particular case, a good matching between the results is obtained for ϵ equal to 1.25 δx. The results are plotted in orange in Fig. 6.13 and 6.14.

At this point, we investigated the behaviour of the solutions with the PCM of the blades. We performed a open-loop control simulation with fixed control torque and with a reference wind velocity of 9 m/s. The results for the standalone FAST simulation and the SOWFA coupled simulation with ϵ equal to 1.25 are reported in Fig. 6.15 and 6.16. The average values show an acceptable difference of about 1%, but it is possible to note a discrepancy in particular in the dynamics of the generated power. This can be further clarified analysing the spectrum of the signal, reported in Fig. 6.17. Both the solutions show an oscillation with the same period of the PCM, but in the standalone FAST simulation also the 3P frequency is particularly excited and a visible peak is present in the spectrum.

A more in-depth analysis of this phenomenon should be performed, taking into account the possible influence of the smearing factor or other numerical parameters. However the lack of data for validation makes this analysis a very challenging task, that is beyond the aims of this work.

This thesis wants to be a preliminary analysis of the effects of PCM control with respect to a standard controller, therefore we are interested in the relative changes caused by the PCM. We performed all the simulations using the same parameters so the comparison is coherent and the results can be trusted for preliminary considerations.

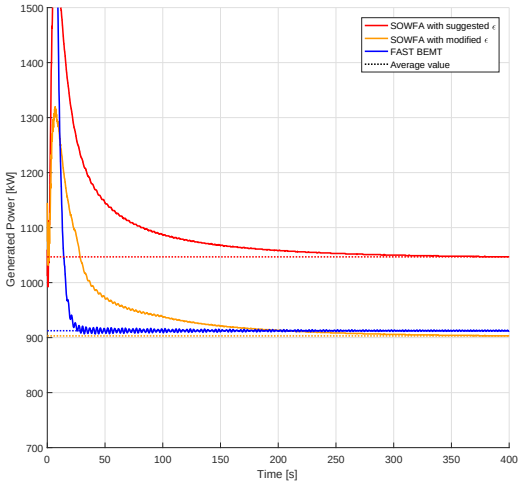


Figure 6.13: Comparison of different simulations in the same trim point – Generated power.

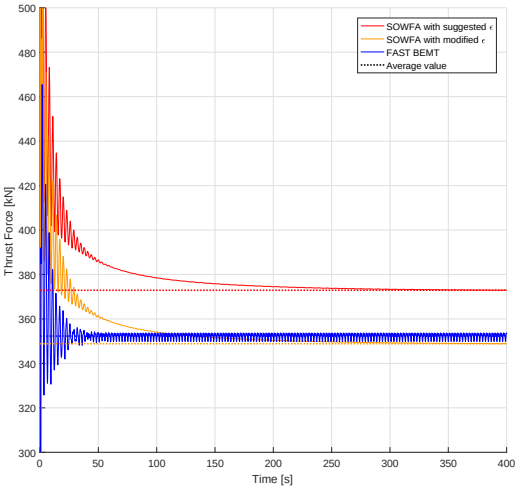


Figure 6.14: Comparison of different simulations in the same trim point – Rotor thrust.

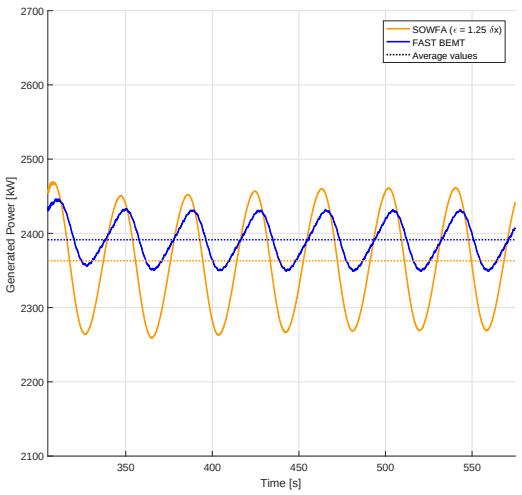


Figure 6.15: Comparison of standalone FAST and SOWFA simulations with PCM – Generated power.

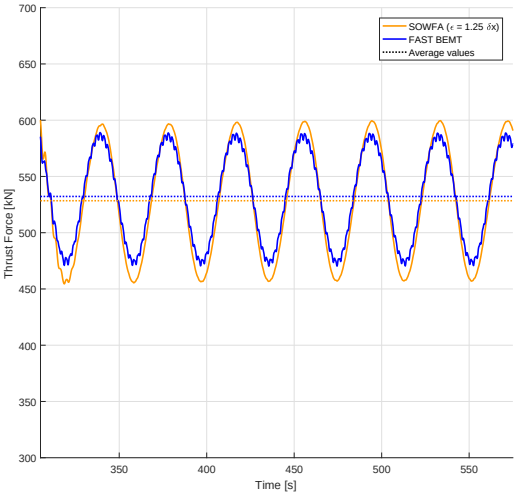


Figure 6.16: Comparison of standalone FAST and SOWFA simulations with PCM – Rotor thrust.

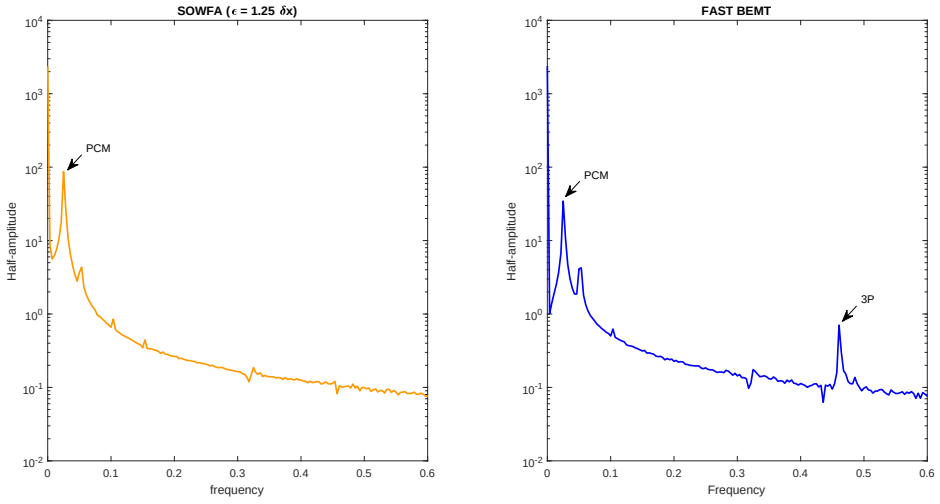


Figure 6.17: Comparison of standalone FAST and SOWFA simulations with PCM – Spectrum of generated power.

6.4 Detailed results for optimal periodic collective motion

In this section we will analyse in detail the effects produced by the periodic collective motion with the optimal parameters found in the previous section, from different points of view. We will report the resulting flow field downstream the turbine and the estimated power production increase, but we will also investigate the effects of the periodic pitch angle variations on the wind turbine structure.

6.4.1 Effects on the flow field

First of all in Fig. 6.18, 6.19 and 6.20 we present the flow average velocity in the y-z plane at distances of 4, 5 and 6 diameters respectively, which are the most relevant for the design of a wind farm.

It is possible to notice that the footprint of the wake is larger for the rotor with PCM implemented, but considering the wind velocity its values are visibly lower for the rotor with the standard controller.

Table 6.1 reports the values of average wake velocity computed using 6.3. The increase in flow velocity is relevant, with values up to 43% higher than the case without PCM.

Distance	**Average wake velocity with optimal PCM** [m/s]	**Average Wake Velocity without PCM** [m/s]	**Velocity deficit reduction** [m/s]
4 Diameters	3.9251	2.7359	1.1892
5 Diameters	4.1552	2.9575	1.1977
6 Diameters	4.3753	3.4013	0.9740

Table 6.1: Average wake velocity – with reference wind velocity of 6.5 m/s.

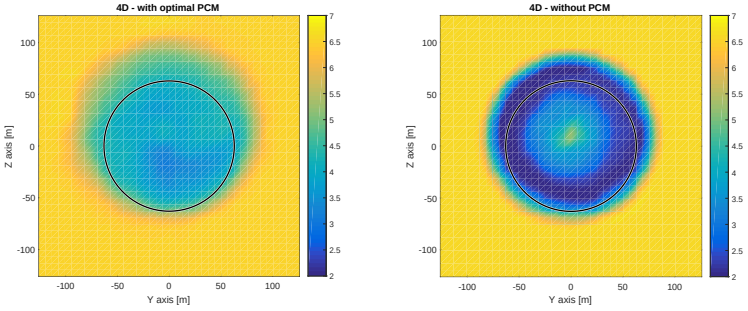


Figure 6.18: Average flow velocity at 4 diameters – with optimal PCM and without PCM.

6.4.2 Effects on power production

In this thesis we simulated a simplified case with only one wind turbine in the flow field, so detailed results about the power output of downstream wind turbines are not available. A first estimation can be obtained using the formula 6.5. Using wake velocity obtained from the simulations we can compute the power generated by a wind turbine installed downstream. Since the power production of the wind turbine goes with the third power of the wind

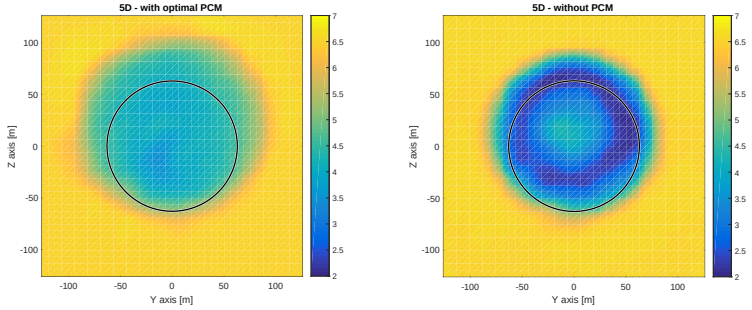


Figure 6.19: Average flow velocity at 5 diameters – with optimal PCM and without PCM.

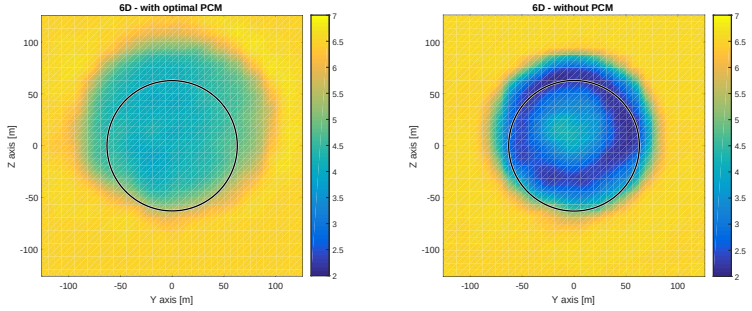


Figure 6.20: Average flow velocity at 6 diameters – with optimal PCM and without PCM.

velocity we can expect large increases in the downstream turbine.

$$P = \frac{1}{2}\rho\, U^3\, A_{rotor}\, C_P \tag{6.5}$$

Analysing the results reported in Tab. 6.1 we can appreciate the effectiveness of the PCM control: when it is not implemented the downstream wind turbine faces a wind that is below the cut-in wind speed. For these particular conditions the downstream wind turbine is switched off and not generating power at all.

We also performed simulations with the same optimal PCM parameters but with a higher wind speed of 9 m/s, to evaluate a condition where the downstream velocity is always above the cut-in wind speed. The resulting average wake velocity is reported in Tab. 6.2.

Distance	**Average wake velocity with optimal PCM** [m/s]	**Average Wake Velocity without PCM** [m/s]	**Velocity deficit reduction** [m/s]
4 Diameters	5.3280	3.9960	1.3320
5 Diameters	5.6557	4.1398	1.5159
6 Diameters	5.8912	4.4600	1.4312

Table 6.2: Average wake velocity – with reference wind velocity of 9 m/s.

Another relevant effect to take into account is the periodic variation of the flow velocity induced by the PCM: the downstream flow velocity oscillates across the average value with approximately the same frequency of the PCM. This behaviour can be observed in Fig. 6.11. A close-up plot of U_{rotor} measured at a distance of 6 diameters with reference wind velocity of 9 m/s is shown in Fig. 6.21, along with the average wake velocity value from Tab. 6.2.

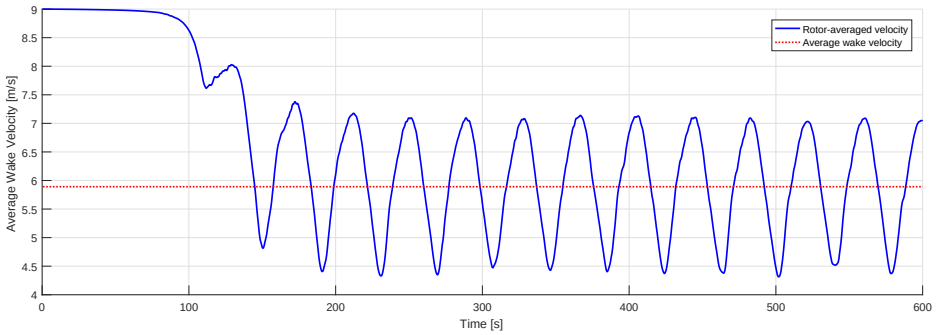


Figure 6.21: Rotor-averaged instantaneous velocity at 6 diameters – values for optimal PCM parameters.

The oscillating behaviour of the flow velocity produces an extra beneficial effect on the power production: since the produced power is proportional to the third power of the velocity, the production will be higher in the case of an oscillating flow field rather than in the case of a constant flow field with the same mean value. This is due to the following property[2] of the average value function $\overline{\cdot}$:

$$\overline{(U^3)} > \left(\overline{U}\right)^3 \tag{6.6}$$

This beneficial effect has already been noticed and small but sensible increases in the power production have been observed.

Substituting property 6.6 in the power production formula 6.5 we can derive the equation used to estimate the average power production $\overline{P}$ of the downstream turbine:

$$\overline{P} = \frac{1}{2}\rho\,\overline{(U(t)^3)}\,A_{rotor}\,C_P \tag{6.7}$$

Using the estimated power production of the downstream turbine and the results from the numerical simulations for the upstream turbine, we are able to compute the overall power increase of the system.

The results are reported in tables 6.3 and 6.4, respectively for 6.5 m/s and 9 m/s wind velocities.

From the results we observe an increase in the overall power production in all the cases. Excluding the two particular cases when the downstream turbine is inactive, the increase ranges from 12 to 15 % which is a positive result, similar to other effective wind farm control techniques. Another positive aspect is the small loss of generated power from the upstream turbine, as we expected. A loss equal to just the 2% is very interesting when compared to other control techniques like yaw misalignment that causes losses up to 10%.

So far we have seen that from the aerodynamic point of view the implementation of the periodic collective motion produces positive effects: the smaller loss of velocity in the downstream flow field directly reflects into a higher power production of the downstream turbine. The production of downstream turbines is still far from ones in the undisturbed wind, but the improvement due to the PCM is significant.

[2] A mathematical proof is presented in appendix C.

[3] Using the value of C_P from the simulations and considering a generator efficiency of 95%.

	Upstream turbine power production [kW]	**Downstream turbine power production**[3] [kW]			**Total power production** [kW]		
		4D	5D	6D	4D	5D	6D
without PCM	857	0	0	121	857	857	978
with PCM	840	199	239	275	1039	1079	1115
Difference	-2 %	–	–	127 %	21 %	26 %	14 %

Table 6.3: Power production of the two turbines system – reference wind velocity of 6.5 m/s.

	Upstream turbine power production [kW]	**Downstream turbine power production**[3] [kW]			**Total power production** [kW]		
		4D	5D	6D	4D	5D	6D
without PCM	2302	203	226	282	2505	2528	2584
with PCM	2266	537	632	695	2803	2898	2961
Increase [%]	-2 %	165 %	180 %	147 %	12 %	15 %	15 %

Table 6.4: Power production of the two turbines system – reference wind velocity of 9 m/s.

6.4.3 Mechanical effects on the wind turbine

It is now important to analyse the loads produced by the implementation of the PCM control and how they affect the mechanical structure of the wind turbine. The results presented in this part of the thesis were computed through FAST simulations with reference wind speed of 6.5 m/s.

Figure 6.22 shows the thrust force acting on the rotor, which is measured at the low-speed shaft, directly connected into the rotor hub.

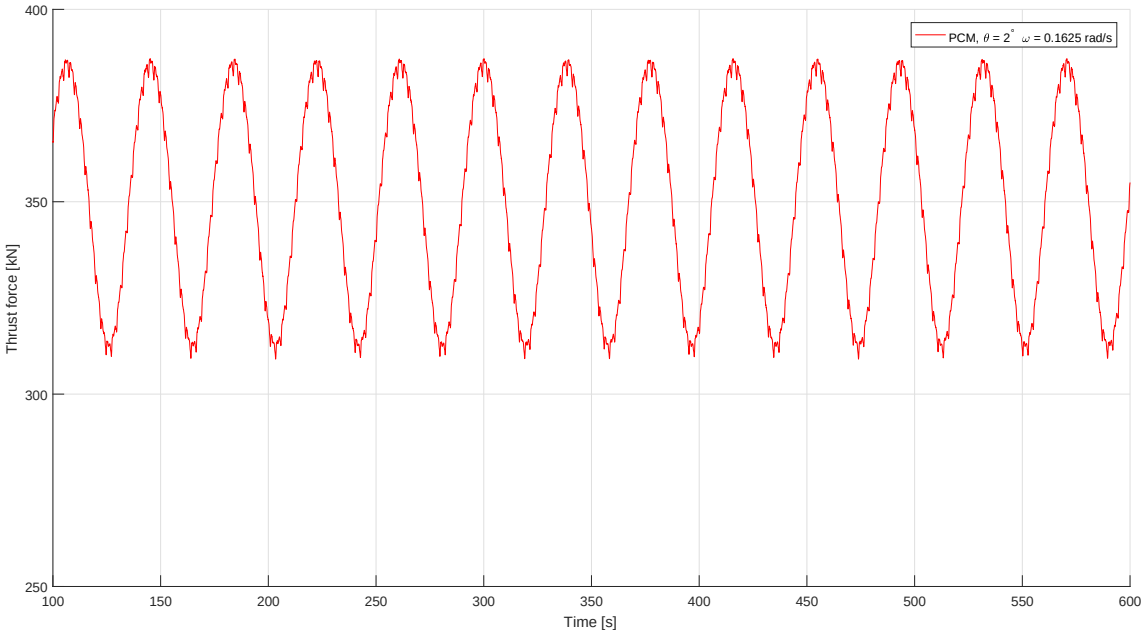


Figure 6.22: Thrust force on the rotor.

Figure 6.23 reports both the torque acting on the rotor, measured at the low-speed shaft,

and the torque acting on the generator, measured at the high-speed shaft. The low-speed shaft and the high-speed shaft are connected through a gearbox with ratio equal to 97.

Figure 6.24 reports the rotor power and the power that is transmitted to the generator. It's is possible to notice that the generator power is slightly lower than the rotor power because we assumed an efficiency of 95% for the gearbox and generator assembly.

Instead of analysing directly the forces, the torque or the power, it is more convenient to compute the thrust, torque and power coefficients. They can be obtained from the definitions:

$$\text{Aerodynamic power:} \qquad P_A = \frac{1}{2}\rho V^3 A C_P \tag{6.8}$$

$$\text{Aerodynamic torque:} \qquad Q_A = \frac{1}{2}\rho V^2 A R C_Q \tag{6.9}$$

$$\text{Aerodynamic thrust:} \qquad T_A = \frac{1}{2}\rho V^2 A C_T \tag{6.10}$$

Since we are not neglecting the rotor moment of inertia, that in our case is equal to 38677040 $kg{\cdot}m^2$, the equilibrium equations for the aerodynamic torque and power are:

$$J\frac{\mathrm{d}\Omega}{\mathrm{d}t} = Q_A - Q_{LSS} \tag{6.11}$$

$$J\Omega\frac{\mathrm{d}\Omega}{\mathrm{d}t} = P_A - P_{LSS} \tag{6.12}$$

where
- J is the rotor moment of inertia
- Ω is the angular speed of the rotor
- Q_{LSS} is the low-speed shaft torque estimated by FAST
- P_{LSS} is the low-speed shaft power estimated by FAST

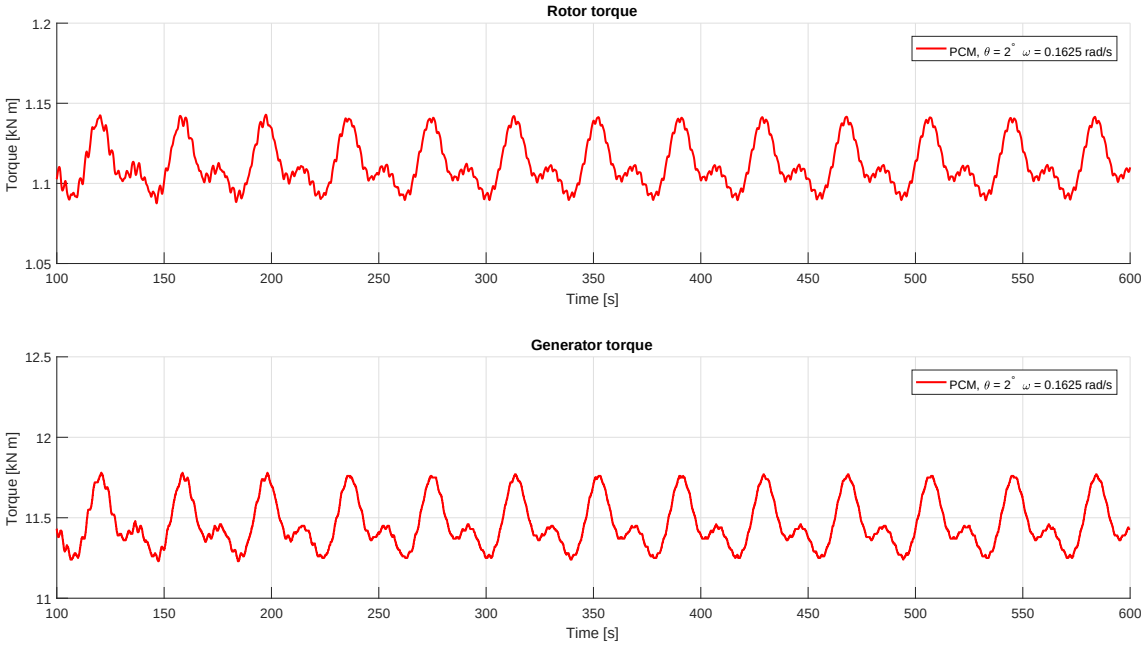


Figure 6.23: Torque on the rotor (upper) and on the generator (lower).

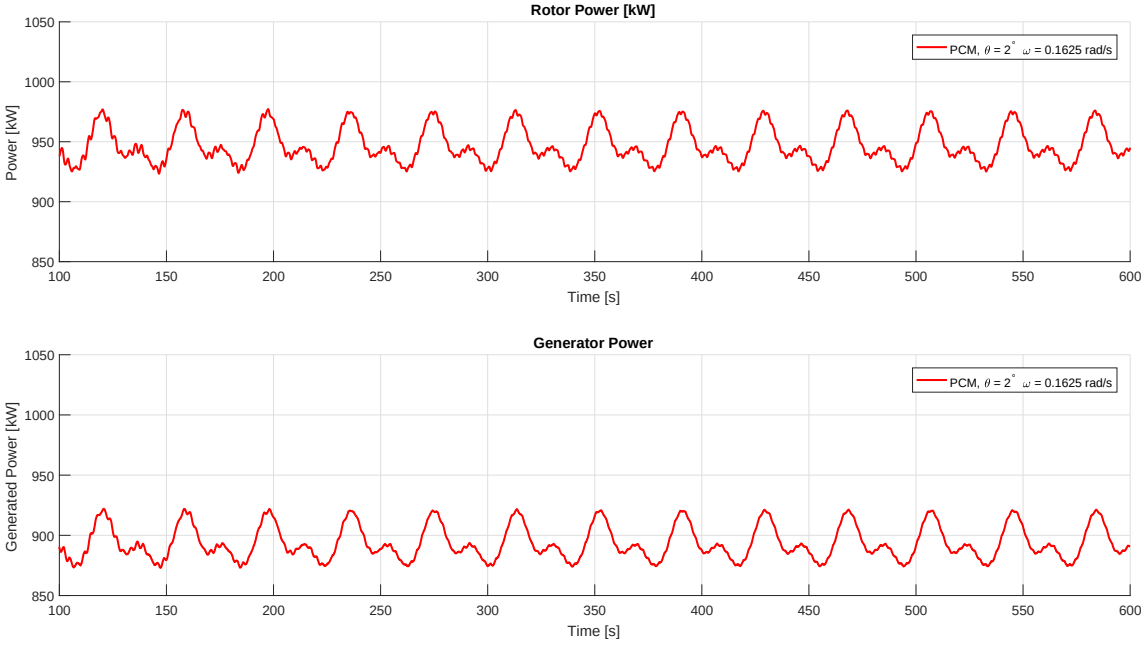


Figure 6.24: Rotor power (upper) and generator power (lower).

Substituting 6.11 into 6.9 and 6.12 into 6.8, the coefficients result to be:

$$\text{Power coefficient:} \qquad C_P = \frac{2(P_{LSS} + J\Omega\frac{\mathrm{d}\Omega}{\mathrm{d}t})}{\rho V^3 A} \tag{6.13}$$

$$\text{Torque coefficient:} \qquad C_Q = \frac{2(Q_{LSS} + J\frac{\mathrm{d}\Omega}{\mathrm{d}t})}{\rho V^2 A R} \tag{6.14}$$

$$\text{Thrust coefficient:} \qquad C_T = \frac{2T_{LSS}}{\rho V^2 A} \tag{6.15}$$

The time history of the coefficients is shown in Fig. 6.25. They have been filtered through a 8th order low-pass Butterworth filter, in order to remove any spurious oscillation.

The power coefficient plot reports also the value of $C_P{}^*$, the optimal value used in region 2 when the pitch angle is fixed. It is possible to note that the values of C_P are on average lower than $C_P{}^*$. That's because with a standard control strategy, the pitch angle in region 2 is fixed and chosen such that it guarantees a value of power coefficient close (ideally equal) to the maximum possible for the blade. On the other hand when the PCM is active the blade pitch angle varies and hence the power coefficient spans a range of values that are on average lower.

From the torque coefficient plot we can see that the torque varies in a periodic way, and this will cause also the rotation speed to vary with the same period.

From the the thrust coefficient plot and the thrust force plot in Fig. 6.23 we can clearly see that the thrust on the rotor varies in a sinusoidal manner, with the same frequency of the PCM. The sinusoidal force transmitted from the rotor to the flow is the key of PCM effectiveness: we induce a pulsation in the flow that interacting with the turbulent structures produces the desired enhancement of the wake mixing.

FAST can also estimate the loads acting on various point of the wind turbine structure. In particular in this thesis we analysed the rotor behaviour, hence we focused on the loads in the crucial locations of the blade root and the hub.

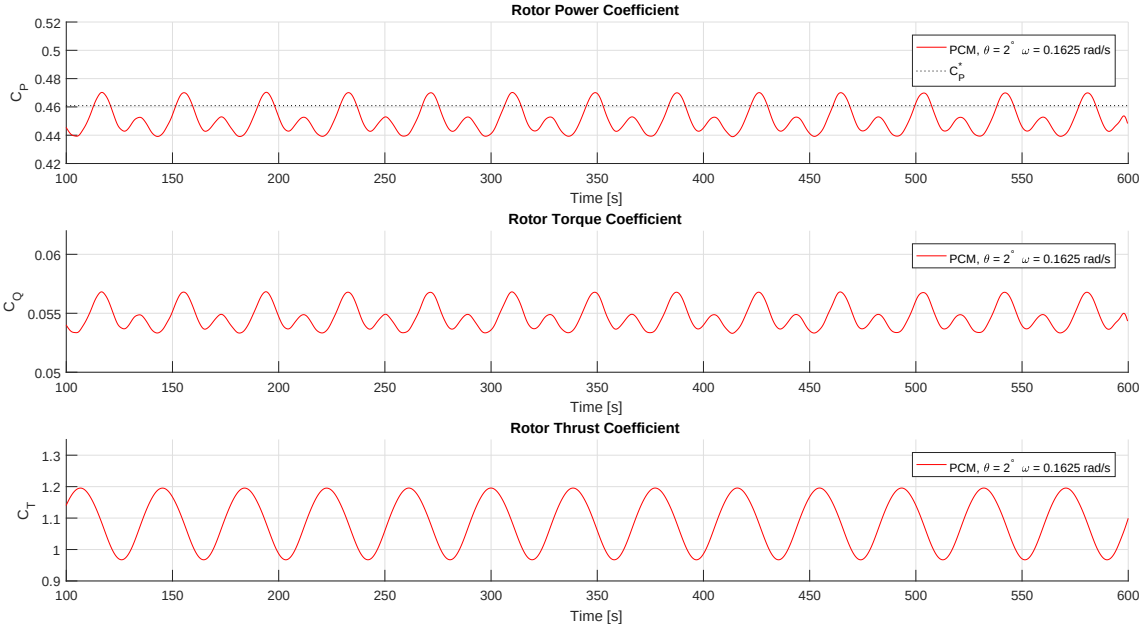


Figure 6.25: Power (upper), torque (middle) and thrust (lower) coefficients.

In general the loadings increase in average value due to the periodic collective motion and they often show a periodic oscillation with the same period of the PCM. The detailed results are reported in appendix B.

To understand whether the loads on the structure affect in a significant way the life cycle of the wind turbine we need a specific indicator.

A common metric for comparing different loads and the fatigue they induce on the structures is the so called Damage Equivalent Load (DEL). This metric has already been used in other similar works, like the paper from Yuan and Tang about load mitigation with an adaptive pitch controller [22].

The damage equivalent load is defined as a constant amplitude load with fixed mean value and frequency, which produces an equivalent damage to the considered variable spectrum loads.

The equation for computing DEL can be obtained from the definition of the short-term fatigue damage D_j caused by a load L_j (Miner's rule):

$$D_j = \sum_i \frac{n_{ji}}{N_{ji}} = \frac{n_j^{eq}}{N_j^{eq}} \tag{6.16}$$

for each time series j, n_{ji} are the stress cycles at stress level i sustained by the structure and N_{ji} is the number of cycles to failure at stress level i. Cf. [23] for a detailed explanation. Consequently, n_j^{eq} are the total equivalent fatigue counts for the j^{th} time series and N_j^{eq} is the equivalent number of cycles until failure. For a DEL they can be defined as:

$$n_j^{eq} = f^{eq} T_j \tag{6.17}$$

$$N_j^{eq} = \left(\frac{L^U - L^M}{\frac{1}{2} DEL_j} \right)^m \tag{6.18}$$

$$\text{where} \quad \begin{cases} f^{eq} \text{ is the DEL frequency} \\ T_j \text{ is the total duration of the time series } j \\ L^U \text{ is the ultimate load} \\ L^M \text{ is the DEL mean value} \\ m \text{ is the Wöhler exponent} \end{cases}$$

Inverting equation 6.18 we can retrieve the formula for computing the damage equivalent load:

$$DEL_j = \left(\frac{\sum_i (n_{ji} (L_{ij}^R)^m)}{n_j^{eq}} \right)^{\frac{1}{m}} \tag{6.19}$$

where L_{ij}^R is the load range for i^{th} stress cycle and j^{th} time-series, it can include the so called Goodman correction, to take into account that every cycle i has a different mean value.

The DELs have been computed for the blade root loads, in particular flapwise and edgewise forces, and the hub loads, in particular the thrust force and the yawing and nodding moments.

The computations have been performed with the aid of $MLife$, a MATLAB-based toolbox developed by the NREL for post-processing the outputs of FAST simulations.

We have decided to compute the DELs at an equivalent frequency of 10 Hz and with zero mean value. The results for the flapwise and edgewise blade root forces are reported in Tab. 6.5 while the results for hub loads are reported in Tab. 6.6. We have decided to include also the DELs for the case when the amplitude of the PCM is equal to 1.5°, in order to evaluate whether a trade-off between optimal performance and lower loadings on the turbine might be an overall better solution.

Simulation	**Flapwise DEL** [kN]	**Edgewise DEL** [kN]
with PCM, $\theta_{PCM} = 2°$	27.5	228
with PCM, $\theta_{PCM} = 1.5°$	25.9	228
without PCM	19.5	228

Table 6.5: Blade root forces damage equivalent loads.

Simulation	**Thrust DEL** [kN]	**Yaw moment DEL** [kNm]	**Nod moment DEL** [kNm]
with PCM, $\theta_{PCM} = 2°$	49.5	745	733
with PCM, $\theta_{PCM} = 1.5°$	38.9	745	730
without PCM	8.47	749	735

Table 6.6: Hub damage equivalent loads.

The results show that the periodic collective motion has a large impact on the flapwise forces, but it does not affect the edgewise forces in a sensible manner. It can be noted that the amplitude of the PCM does affect the loads on the blade. The thrust force on the hub is severely affected by the PCM, also showing a sensible difference between the case with $\theta_{PCM} = 2°$ and the sub-optimal case with $\theta_{PCM} = 1.5°$. The yawing and nodding moment show very small variations due to PCM, but overall they can be considered unaffected by the PCM.

To better understand the increase caused by PCM, the DELs are normalized taking the case without PCM control as reference. The flapwise normalized DELs are plotted as bar

chart in Fig. 6.26 and the thrust normalized DELs are plotted in Fig. 6.27. The relative increases caused by the PCM control are reported in tables 6.7 and 6.8.

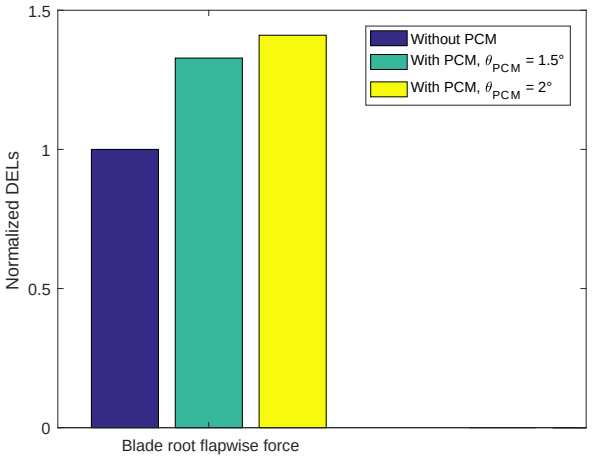


Figure 6.26: Normalized flapwise damage equivalent loads – bar chart.

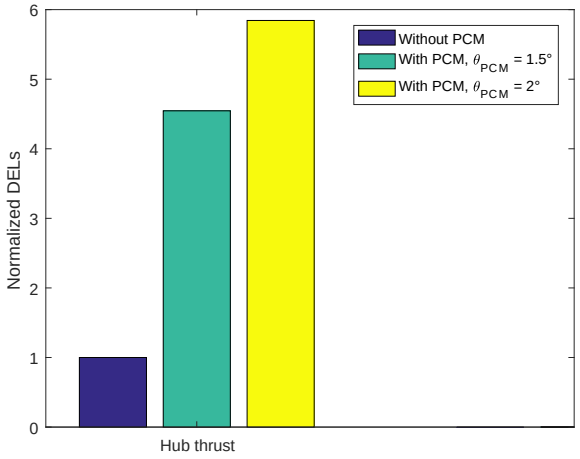


Figure 6.27: Normalized thrust damage equivalent loads – bar chart.

These results, especially the loads on the hub caused by the thrust force, may look critical for the wind turbine structure and its life cycle. But before drawing any conclusion we need to contextualize them.

In fact during the operational life of a wind turbine, the PCM will be enabled just for a limited amount of time: as already explained in the previous chapters, the PCM is unnecessary when the turbine is working in region 3. Furthermore PCM needs to be turned on when the wind direction is such that the wind turbines are aligned, but in well-design wind farms this is an unlikely condition.

In order to produce a realistic estimation of the impact of DELs increase in the wind turbine life cycle, we considered a real world application.

Simulation	Normalized Flapwise DEL	Increase due to PCM
with PCM, $\theta_{PCM} = 2°$	1.41	+41 %
with PCM, $\theta_{PCM} = 1.5°$	1.33	+33 %
without PCM	1	–

Table 6.7: Normalized flapwise damage equivalent loads – relative increases.

Simulation	Normalized Flapwise DEL	Increase due to PCM
with PCM, $\theta_{PCM} = 2°$	5.84	+484 %
with PCM, $\theta_{PCM} = 1.5°$	4.54	+354 %
without PCM	1	–

Table 6.8: Normalized thrust damage equivalent loads – relative increases.

The following assumptions on wind conditions and on the wind farm layout are based on the NORCOWE reference wind farm for the $FINO3$ zone. $FINO3$ refers to a met mast installed in the North Sea, 80 km west of the island of Sylt. The data gathered by the met mast was used to produce the wind rose reported in Fig. 6.28 and then to elaborate an optimal design for an off-shore wind farm. The wind rose gives the probability of occurrence as a function of both the velocity and the direction of the wind. The wind velocity distribution can also be approximated with a Weibull distribution. Gryning et al. interpolated the data from the wind mast fo find the best fitting parameters for the Weibull distribution [24]. The resulting distribution is reported in Fig. 6.29. The wind turbines are installed along -15° direction (starting from North, positive clockwise), as it has been identified as the direction with lowest probability of wind. Figure 6.30 shows the layout of a simple two turbines wind farm. Considering the distance between the two wind turbines to be equal to 4 diameters (504 m), the wake of the upstream turbine will hit the downstream rotor for winds that are directed from -45° to -15°. Since from previous simulations we learned that the wake spreads in the downstream direction, it would be safer to activate the PCM for a wider range of wind directions, for example from -55° to -5°.

In terms of wind velocity we investigated a neighbourhood of 6.5 m/s, in particular the range from 6 m/s to 7 m/s.

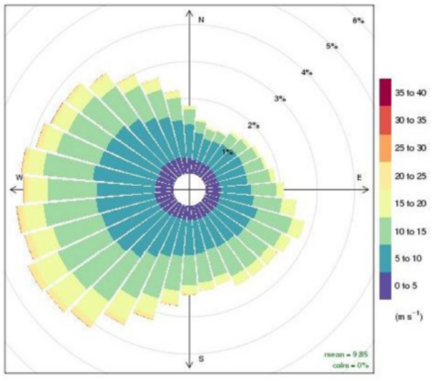


Figure 6.28: $FINO3$ wind rose.

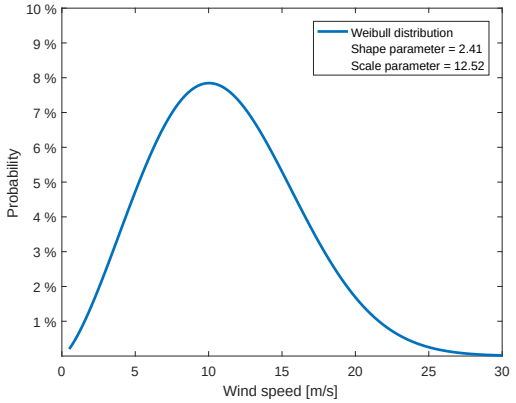


Figure 6.29: $FINO3$ wind speed distribution.

So far we have defined the range of PCM activation in terms of wind velocity and direction.

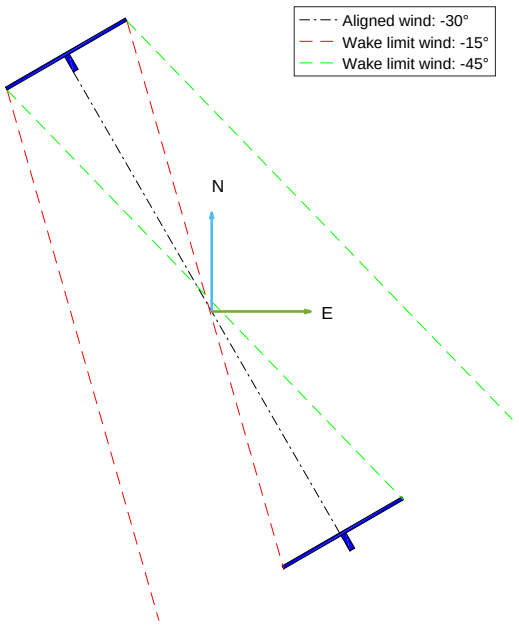


Figure 6.30: Wind farm layout.

We have identified two possible choices of the activation range: a "Minimal" region and a "Safe" region, wider than the previous in terms of wind direction. A recap of the two ranges is reported in Tab. 6.9.

Correlating the probability distribution given by the Weibull distribution and the wind rose with wind spectra for the activation of the PCM, we can obtain the overall probability that the PCM control is active and working on the wind turbine. The resulting probabilities are reported in the last column of Tab. 6.9.

	Direction range	**Velocity range**	**Probability**
"Minimal" region	-45° < Ψ <-15°	6 m/s < U < 7 m/s	0.45 %
"Safe" region	-55° < Ψ <-5°	6 m/s < U < 7 m/s	0.87 %

Table 6.9: PCM activation regions.

Now that we have an idea of the incidence of the PCM control on the wind turbine operating cycle, we can weight the damage equivalent loads with their occurrence probability, in order to obtain an indicator of the impact of the PCM.

The results for the blade root flapwise force and the hub thrust DELs with the two possible activation intervals are presented in tables 6.10 and 6.11.

These results can be interpreted as how much the loads due only to the periodic collective motion affect the blade's root and the hub during their entire lifetime.

The incidence of PCM induce loads on the the blade root is not particularly high, but we have to bear in mind that this value is referred to a very narrow range of wind velocities. If we hypothetically assume that the DEL increase is constant with the wind velocity and compute the impact considering all the region $1\frac{1}{2}$ and region 2 winds, we end up with a

		Flapwise DEL increase due to PCM	
		$\theta_{PCM} = 1.5°$ ΔDEL = +33 %	$\theta_{PCM} = 2°$ ΔDEL = +41 %
"Minimal" region	P = 0.45 %	+ 0.15 %	+ 0.18 %
"Safe" region	P = 0.87 %	+ 0.29 %	+ 0.38 %

Table 6.10: Lifetime equivalent impact of PCM control at 6.5 m/s wind speed on blade root loads.

		Thrust DEL increase due to PCM	
		$\theta_{PCM} = 1.5°$ ΔDEL = +354 %	$\theta_{PCM} = 2°$ ΔDEL = +484 %
"Minimal" region	P = 0.45 %	+ 1.59 %	+ 2.18 %
"Safe" region	P = 0.87 %	+ 3.08 %	+ 4.21 %

Table 6.11: Lifetime equivalent impact of PCM control at 6.5 m/s wind speed on hub thrust force.

lifetime equivalent increase in the order of +2/4 %. Further expanding the PCM activation region to wind above the rated speed will produce even higher increases. Since the blades are at high risk of fatigue damage these increases in DELs can not be neglected.

The hub is also affected by the PCM induced fatigue loads, with a very high impact over its lifetime. However the hub is mechanical element that is less prone to fatigue damage and so the increase in the loads might not be as critical.

Anyhow it appears that a wide use of PCM control might be inconvenient, as the higher power production is counterbalanced by a higher wear of the mechanical parts as well as a shorter service life of the blades. A more conservative application of PCM control, for example reducing the activation range only to particularly critical wind conditions, seems to be a better approach.

Since wind turbines are complex mechanical assemblies, the estimation of the life cycle, the installation and the maintenance cost are very difficult tasks and they're specific to the particular wind turbine and wind farm considered. This kind of analysis is beyond the aims of this thesis.

Chapter 7

Conclusions

The work done in this thesis was focused on demonstrating the effectiveness of PCM in terms of wake deficit and power production. We extended the analysis to the structural field, in order to produce a first estimation of the cost of PCM implementation in terms of induced loads on the structure.

To perform numerical simulations we had to set up a new virtual environment based on the CFD software SOWFA coupled with the aero-elastic analysis tool FAST. We met the objective of providing a working CFD analysis environment to the POLI-Wind research group: SOWFA and FAST are correctly installed on MARCONI HPC and ready to use.

We performed numerical simulations with different values for the PCM control parameters. The results we obtained are in agreement with the current literature. In fact we demonstrated that it is possible to achieve an increment in the range of 12 - 15 % in the overall power producion using the PCM control technique. On the other hand we also noted an increase in the loads on the wind turbine blades. Using the Damage Equivalent Loads in the upstream turbine (i.e. the one performing the PCM) as metric for comparing the effects of the loads on the structure, we measured an increment up to 44 % in the fatigue loads on the blades. Computing the equivalent impact of this loads over the blade lifetime we obtained increments from 2 to 4%, depending on how the PCM is implemented.

The conclusion we can draw from this preliminary study is that the design of a controller implementing PCM must be a trade-off between higher power production and induced loadings. Certainly PCM can exploit the wake meandering mechanism to produce a faster wake, but its activation conditions must be carefully evaluated in order to avoid significant effects on the wind turbine loading status.

A possible optimal wind farm controller might include different control techniques, switching to PCM control only when other strategies, like adaptive de-rating or wake redirection by yawing, would be too expensive in terms of loss of power production in the upstream turbines.

Outlook

Starting from the work done in this thesis there are many possible future developments:

- LES of more realistic scenarios. An interesting case is including multiple wind turbines to directly measure the effects of PCM on downstream turbines. Another possible case is imposing a more complex flow field, for example simulating the atmospheric boundary layer in different stability states or including wind veering.

- LES of a wind tunnel experiment, in order to validate the numerical results.

- Try to reduce the upstream turbine power production loss using a different control strategy, that acts both on the blade pitch angle and the demanded torque.

- Evaluate the effects of a cyclic variation of the pitch angle on the wake meandering and investigate whether it can be used to further enhance the wake mixing.

- Integrate the PCM control with other wind farm control strategies, e.g wake redirection.

Appendix A

Installation of SOWFA and FAST on Marconi

SOWFA requires a working installation of OpenFOAM to work correctly, so the install procedure will follow this steps:

1. Installation of OpenFOAM 2.4.x
2. Installation of FAST v8
3. Installation of SOWFA

A.1 Installation of OpenFOAM 2.4.x

Create the install folder in the user home directory

```
cd ~
mkdir OpenFOAM
cd OpenFOAM
```

Download OpenFOAM source code from GitHub:

```
git clone https://github.com/OpenFOAM/OpenFOAM-2.4.x.git
git clone https://github.com/OpenFOAM/ThirdParty-2.4.x.git
```

Download and extract the source code of the required packages :

```
cd ThirdParty-2.4.x
mkdir download
wget -P download http://www.paraview.org/files/v4.1/ ...
                  ParaView-v4.1.0-source.tar.gz
wget -P download  https://gforge.inria.fr/frs/download.php/file/ ...
                34099/scotch_6.0.3.tar.gz
wget -P download https://gforge.inria.fr/frs/download.php/file/ ...
                34703/CGAL-4.6.tar.bz2
wget -P download http://sourceforge.net/projects/boost/files/boost/ ...
                1.55.0/boost_1_55_0.tar.bz2
tar -xzf download/ParaView-v4.1.0-source.tar.gz
tar -xzf download/scotch_6.0.3.tar.gz
tar -xjf download/CGAL-4.6.tar.bz2
tar -xjf download/boost_1_55_0.tar.bz2
```

Set the correct version of boost:

```
sed -i -e 's=boost-system=boost_1_55_0=' ...
                OpenFOAM-2.4.x/etc/config/CGAL.sh
```

GMP is not available on MARCONI so we need to download the correct version from the website (https://gmplib.org/list-archives/gmp-announce/2016-December/000047.html), copy it in the user home folder and then compile it:

```
cd ~
tar xvjf gmp-6.1.2.tar.bz2
cd gmp-6.1.2
./configure --prefix=$HOME/OpenFOAM/ThirdParty-2.4.x/ ...
                platforms/linux64/gmp-system/
make
make check
make install
```

The same must be done for mpfr (https://www.mpfr.org/mpfr-current/):

```
cd ~
unzip mpfr-4.0.1.zip
cd mpfr-4.0.1
./configure --prefix=$HOME/OpenFOAM/ThirdParty-2.4.x/ ...
  platforms/linux64/mpfr-system/ --with-gmp-build=$HOME/gmp-6.1.2
make
make check
make install
```

Now we source OpenFOAM:

```
source $HOME/OpenFOAM/OpenFOAM-2.4.x/ ...
                etc/bashrc WM_NCOMPPROCS=4 WM_MPLIB=SYSTEMOPENMPI
```

We can create an alias in the .bashrc file:

```
echo "alias of24x='module load mpi/openmpi-x86_64; source \$HOME/ ...
  OpenFOAM/OpenFOAM-2.4.x/etc/bashrc $FOAM_SETTINGS'" >> $HOME/.bashrc
```

Now we can compile the third party files. First of all we have to compile cmake, MARCONI has cmake available as module but it is a recent version not compatible with OpenFOAM 2.4.x.

```
cd $WM_THIRD_PARTY_DIR
./makeCmake > mkcmake.log 2>&1
wmSET $FOAM_SETTINGS
```

Compile Qt:

```
wget https://download.qt.io/archive/qt/4.7/ ...
                qt-everywhere-opensource-src-4.7.3.zip
unzip qt-everywhere-opensource-src-4.7.3.zip
./makeQt
```

Compile CGAL:

```
cd $WM_THIRD_PARTY_DIR
sed -i -e 's=boost-system=boost_1_54_0=' makeCGAL
./makeCGAL
```

Now that all the prerequisites are install we can compile OpenFOAM:

```
cd $WM_PROJECT_DIR
./Allwmake > log.make 2>&1
```

OpenFOAM has some problems in creating all its folder on MARCONI, so we have to create them manually:

```
cd ~
cd OpenFOAM
mkdir scacciol-2.4.x
```

```
cd scacciol-2.4.x
mkdir run
mkdir platforms
cd platforms
mkdir linux64GccDPOpt
cd linux64GccDPOpt
mkdir bin
mkdir lib
```

OpenFOAM is installed.

A.2 Installation of FAST v8

Before installing FAST, check that all the prerequisites are satisfied:

- OpenFOAM installed in /OpenFOAM/OpenFOAM-2.4.x/ (done in A.1)

- the following modules loaded:
  gnu/6.1.0
  openmpi/1-10.7–gnu–6.1.0
  zlib/1.2.8–gnu–6.1.0
  python/3.5.2
  lapack/3.6.1–gnu–6.1.0
  blas/3.6.0–gnu–6.1.0

Download FAST v8 source code:

```
cd OpenFOAM/scacciol-2.4.x/
git init
git clone https://github.com/gantech/fastv8DriverProgram.git
```

Select the FAST v8 branch that supports SOWFA/OpenFOAM coupling:

```
cd fastv8DriverProgram
git checkout OpenFOAMcoupling
```

Modify the first lines of .sourceMods.sh according to the modules we have loaded:
module unload
module load profile/advanced
module load autoload
if ["${COMPILER}" == 'gnu']; then
module load gnu/6.1.0 cmake/3.5.2 openmpi/1-10.7–gnu–6.1.0
Modify the following lines install.sh to use the newest version of hdf5:

Line 108	from:	wget –no-check-certificate https:// ... /src/hdf5-1.8.18.tar.bz2 &> log.wget
	to:	wget –no-check-certificate https:// ... /src/hdf5-1.8.20.tar.bz2 &> log.wget
Line 111	from:	tar -jxf hdf5-1.8.18.tar.bz2 &> log.untar
	to:	tar -jxf hdf5-1.8.20.tar.bz2 &> log.untar
Line 112	from:	cd hdf5-1.8.18
	to:	cd hdf5-1.8.20

Uncomment in install.sh line number 174: *compileYAMLcpp* in order to compile also YAML.
Now we can compile FAST:

```
COMPILER=gnu BUILD=release LAPACK=lapack ./install.sh
```

Some warning messages may appear but if all the steps print ... $PASSED$ in the terminal output the installation is successful.

A.3 Installation of SOWFA

Before proceeding we have to define the following environment variables, in our case we added these lines to /.bash_profile:

```
FAST_DIR=$HOME/OpenFOAM/scacciol-2.4.x/fastv8DriverProgram/fastv8/
export FAST_DIR
HDF5_DIR=$FAST_DIR/Source/dependencies/hdf5-1.8.20/src/
export HDF5_DIR
```

We have to log-off and log-on to activate the environment variables.
Now we can proceed with the installation of SOWFA, first we download the source code:

```
of24x
cd OpenFOAM/scacciol-2.4.x/
git clone https://github.com/NREL/SOWFA.git
```

Before compiling we have to fix some source code. Navigate to the windEnergy incompressible solvers folder:

```
cd ./SOWFA/applications/solvers/incompressible/windEnergy/
```

We have to edit the file options (found in $SOLVER_FOLDER$/Make/options) in this way:

replace: -I$(WM_PROJECT_USER_DIR)/src/...
with: -I$(WM_PROJECT_USER_DIR)/SOWFA/src/...

For all the following solvers:

- pisoFoamTurbine.ADM
- pisoFoamTurbine.ALMAdvanced
- pisoFoamTurbine.ALM
- pisoFoamTurbine.ALMAdvancedFASTv8
- turbineTestHarness.ALM
- turbineTestHarness.ALMAdvanced
- windPlantSolver.ADM
- windPlantSolver.ALMAdvanced (two times)
- windPlantSolver.ALM
- windPlantSolver.ALMAdvancedFASTv8 (two times)

For the solvers coupled with FAST v8:

- pisoFoamTurbine.ALMAdvancedFASTv8
- windPlantSolver.ALMAdvancedFASTv8 (two times)

We have to edit also these lines in the options file:

replace: -L$(FAST_DIR)/lib \
-lFAST_glin64_release \
-lFAST_cInterface_glin64_release \
-lmap-1.20.10 \
with: -L$(FAST_DIR)/lib \
-lFAST_glin64_release \
-lFAST_cInterface_glin64_release \
-lmap-1.20.10 \
-L$(FAST_DIR)/lib64 \
-llapack \
-lblas \
-llapacke \

We have to create manually some missing symbolic links:

```
cd $FOAM_USER_LIBBIN
ln -s /.../OpenFOAM/scacciol-2.4.x/fastv8DriverProgram/fastv8/...
      lib64/libblas.so libblas.so
ln -s /.../OpenFOAM/scacciol-2.4.x/fastv8DriverProgram/fastv8/...
```

```
        lib64/libblas.so.3 libblas.so.3
ln -s /  ... /OpenFOAM/scacciol-2.4.x/fastv8DriverProgram/fastv8/ ...
        lib64/liblapacke.so liblapacke.so
ln -s /  ... /OpenFOAM/scacciol-2.4.x/fastv8DriverProgram/fastv8/ ...
        lib64/liblapack.so liblapack.so
ln -s /  ... /OpenFOAM/scacciol-2.4.x/fastv8DriverProgram/fastv8/ ...
        lib64/liblapack.so.3 liblapack.so.3
ln -s /  ... /OpenFOAM/scacciol-2.4.x/fastv8DriverProgram/fastv8/ ...
        lib/libmap-1.20.10.so libmap-1.20.10.so
ln -s /  ... /OpenFOAM/scacciol-2.4.x/fastv8DriverProgram/fastv8/ ...
        lib/libFAST_glin64_release.so libFAST_glin64_release.so
ln -s /  ... /OpenFOAM/scacciol-2.4.x/fastv8DriverProgram/fastv8/ ...
        lib/libFAST_cInterface_glin64_release.so ...
        libFAST_cInterface_glin64_release.so
```

It is recommended to create a working folder outside the installation folder but symbolically linked to it, so when SOWFA is updated from GitHub we won't need to update the files in the working folder.
This operation is done by the *makeNewWorkingDir* script. We can edit it to create as many folders as we desire. In our case we want to create a working folder in MARCONI $WORK folder since it has no storage limits, the *makeNewWorkingDir* script will look like this:

Edit this as needed (paths may differ from system to system)
SOWFAdir=$HOME/OpenFOAM/scacciol-2.4.x/SOWFA
Do the recursive copy that symbolically links the files to the source.
cp -rs $SOWFAdir $CINECA_SCRATCH/WORK/
echo "SOWFA compilation directory created in $SOWFAdir"

To create the working folder we just execute the script:

```
./makeNewWorkingDir
```

Finally we can compile SOWFA:

```
./Allwclean
./Allwmake
```

If the compilations ends without critical errors SOWFA is successfully installed.

Appendix B

Loads on the wind turbine rotor used for the computation of DELs

Due to the periodic nature of the loads that wind turbines are subjected to, they are particularly prone to fatigue damage. In particular the blades are highly susceptible to this kind of risk. A in-depth analysis of the fatigue loading in the blades is beyond the aims of this thesis, however we wanted to compute a rough estimation of the loads caused by PCM. They add up to the loads that typically act on the wind turbine blades, which are:

- Gravitational loads due to the mass of the blades
- Centrifugal loads due to the rotation
- Thrust force caused by the wind
- Forces caused by the turbulence

We want to estimate what is the contribution of PCM with respect to the other fatigue loads. The metric chosen for measuring the fatigue loads is the *Damage Equivalent Load (DEL)*. In order to compute it we need the time history of the load that we want to analyse. In our simulations we considered only the rotor of the wind turbine, so the elements we will analyse are the blades and the hub. The time history of the blade root forces for the case without PCM control are reported in Fig. B.1. We can observe that both the flapwise and the edgewise loads oscillate according to the 1P frequency (the one given by the rotational speed). The same loads when the PCM control is implemented are reported in Fig. B.2. Comparing the two figures wa can conclude that the edgewise loads are not affected by the PCM control. The flapwise loads on the other hand are affected by the PCM control in a visible way. On the plot of flapwise loads in Fig. B.2 now we can clearly identify two harmonics: the 1P harmonic and a new slower harmonic with the same period of the PCM ($T = \frac{2\pi}{0.1625\,rad/s} \sim 38$ s).

The loads acting on the rotor hub are measured at the low-speed shaft and take into account the rotation of the hub. Figure B.3 reports the loads when the PCM is not implemented while Fig. B.4 reports the loads when the PCM control is active.

The behaviour of hub loads is similar to the blade loads one: the yawing and the nodding moment are not affected by the PCM, while the thrust force, initially constant, varies with the same frequency of the PCM when it is implemented.

Since the blade root flapwise force and the thrust force on the hub are affect by the PCM in terms of both frequency content and amplitude, we can expect an increase in *Damage Equivalent Loads*. The remaining loads on the other hands should not produce greater DELs.

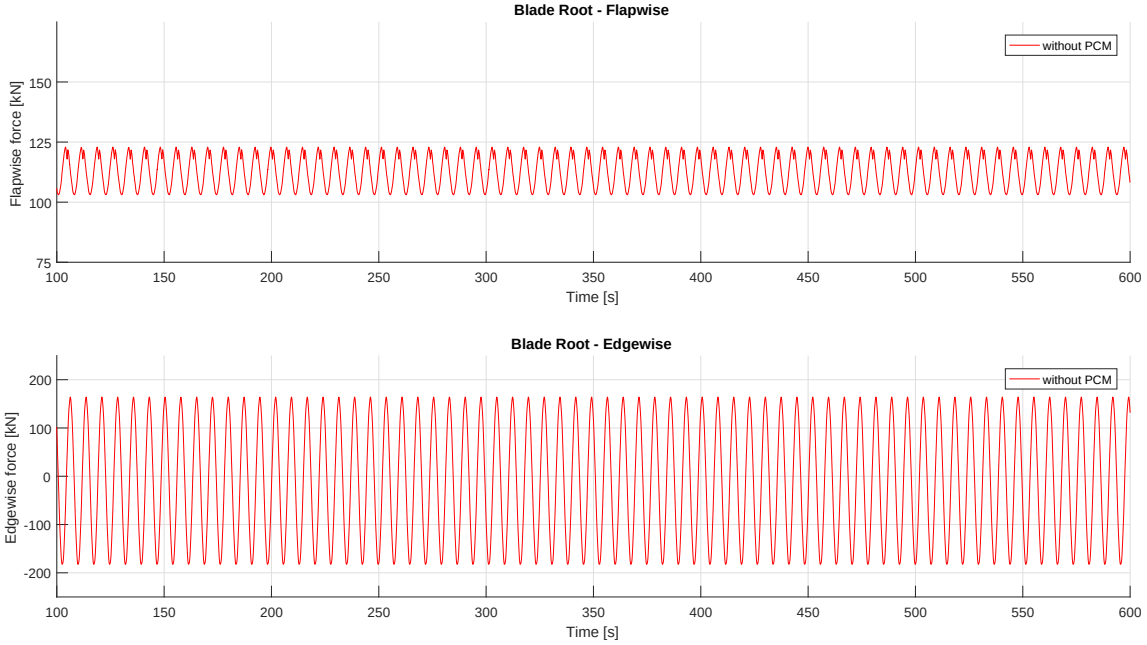


Figure B.1: Blade root forces – without PCM control.

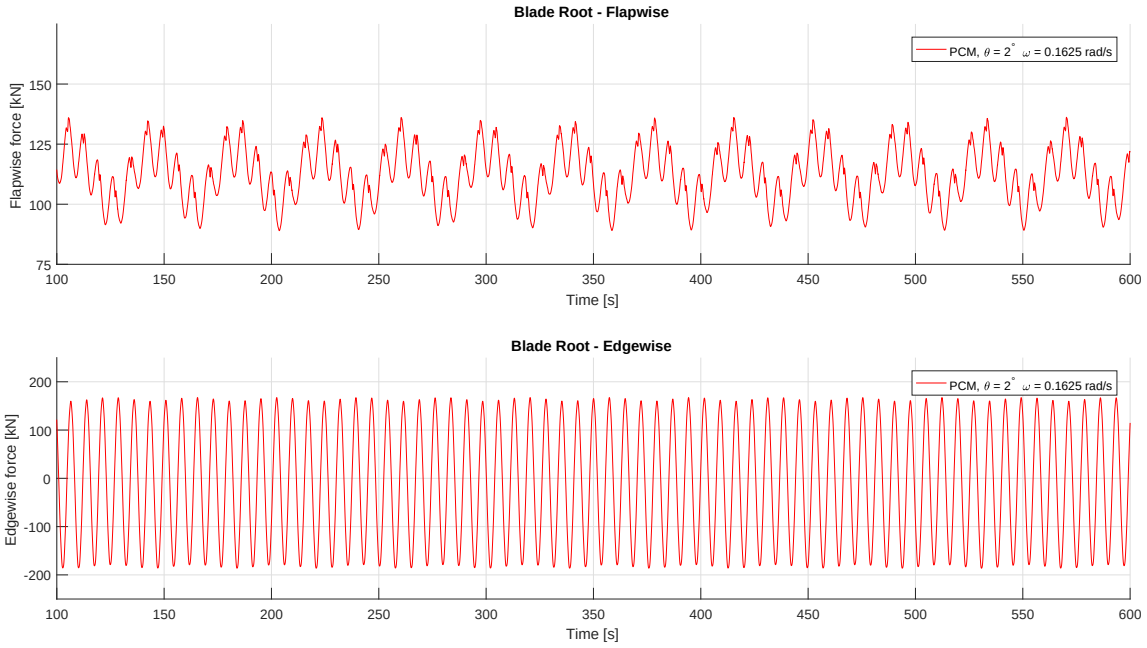


Figure B.2: Blade root forces – with PCM control.

The resulting DELs are reported in chapter 6.4.3.

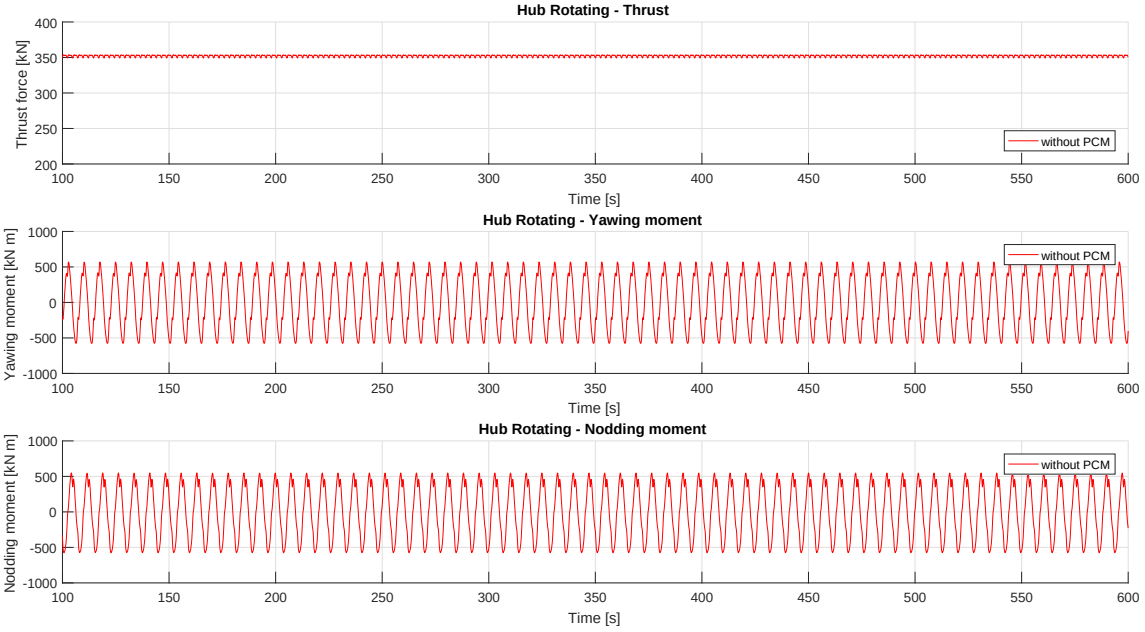


Figure B.3: Hub forces – without PCM control.

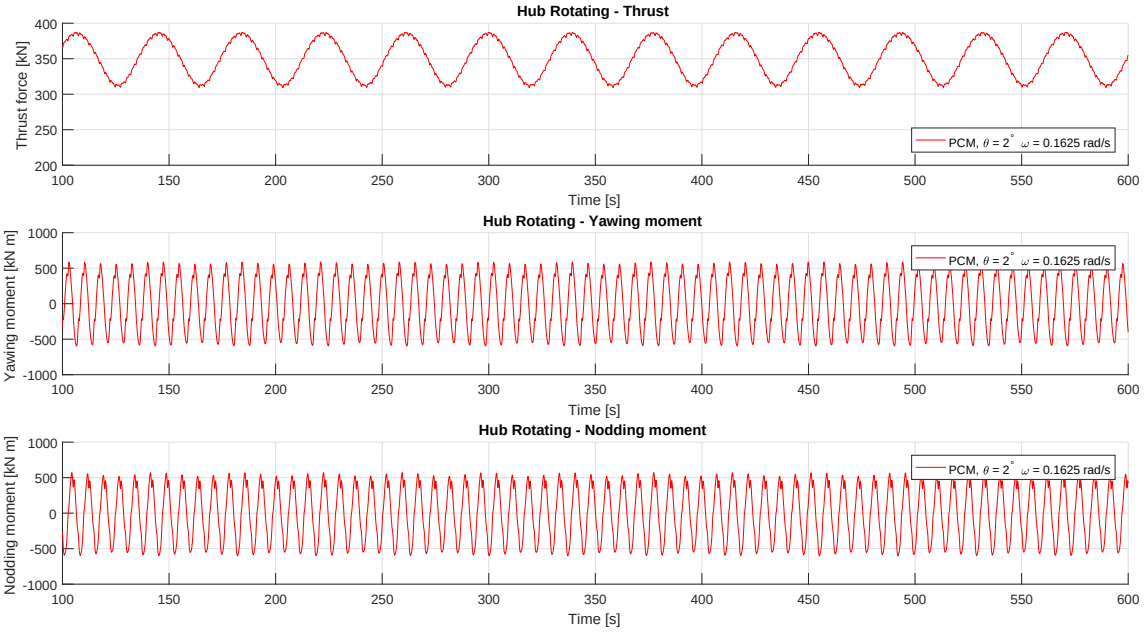


Figure B.4: Hub forces – with PCM control.

Appendix C

Effects of an oscillating flow field on the power production

When considering an oscillating flow U with positive mean value it satisfies the following property:

$$\overline{(U^3)} > (\,\overline{U}\,)^3 \tag{C.1}$$

Let's consider a simple harmonic flow:

$$U(t) = U_0 + u\sin(\omega t) \quad \text{with} \quad U_0 > 0$$

The average value function $\overline{\cdot}$ is defined as:

$$\overline{\cdot} = \frac{1}{T}\int_{\tau}^{\tau+T} \cdot\, dt \quad \text{where T is the period of oscillation} \quad T = \frac{1}{\omega}$$

Computing the right hand side term of C.1 is straightforward:

$$\begin{aligned}
(\,\overline{U(t)}\,)^3 =& \left(\frac{1}{T}\int_{\tau}^{\tau+T} U(t)\,dt\right)^3 \\
=& \left(\frac{1}{T}\int_{\tau}^{\tau+T} U_0 + u\sin(\omega t)\,dt\right)^3 \\
=& \left(\frac{1}{T}\int_{\tau}^{\tau+T} U_0\,dt + \frac{1}{T}\int_{\tau}^{\tau+T} u\sin(\omega t)\,dt\right)^3
\end{aligned}$$

Due to the symmetry of the sine function, its integral over the period T is null. Thereby:

$$(\,\overline{U(t)}\,)^3 = \left(\frac{1}{T}U_o T\right)^3 = {U_0}^3$$

The left hand side of C.1 is equal to:

$$\begin{aligned}
\overline{U(t)^3} =& \frac{1}{T}\int_{\tau}^{\tau+T} U(t)^3\,dt \\
=& \frac{1}{T}\int_{\tau}^{\tau+T} \left(U_0 + u\sin(\omega t)\right)^3 dt \\
=& \frac{1}{T}\int_{\tau}^{\tau+T} \left({U_0}^3 + 3{U_0}^2 u\sin(\omega t) + 3U_0 u^2\sin^2(\omega t) + u^3\sin^3(\omega t)\right) dt \\
=& \frac{1}{T}\int_{\tau}^{\tau+T} {U_0}^3\,dt + \frac{1}{T}\int_{\tau}^{\tau+T} 3{U_0}^2 u\sin(\omega t)\,dt + \frac{1}{T}\int_{\tau}^{\tau+T} 3U_0 u^2\sin^2(\omega t)\,dt + \\
&+ \frac{1}{T}\int_{\tau}^{\tau+T} u^3\sin^3(\omega t)\,dt
\end{aligned}$$

Here the symmetry cancels out the second and the fourth term, but the term proportional to $\sin^2$ does not cancel out.
So equation C.1 can be rewritten as:

$$U_0{}^3 + \frac{1}{T}\int_{\tau}^{\tau+T} 3U_0 u^2 \sin^2(\omega t)\, dt > U_0{}^3 \tag{C.2}$$

Since we assumed a positive mean value ($U_0 > 0$) the contribution of the integral at the left hand side is always positive. We proved C.1.

Let's make an example to further clarify this property. Figure C.5 shows the behaviour of $U(t)$ assuming the following values for the harmonic motion parameters:

$$\begin{cases} U_0 = 2\, m/s \\ u = 1\, m/s \\ \omega = \frac{2\pi}{5}\, rad/s \end{cases}$$

The time history of $U(t)^3$ is reported on the bottom plot along with the values of the two averages. It is immediate to notice that $\overline{(U^3)}$ (the magenta dashed line) and $\left(\overline{U}\right)^3$ (the solid red line) yield two different values, as predicted. In particular:

$$\begin{cases} \overline{(U^3)} = 11\, m/s \\ \left(\overline{U}\right)^3 = 8\, m/s \end{cases}$$

Looking back at equation C.2 we can observe that for a non oscillating velocity (u=0) the left hand side and the right hand side are equal. So only for non oscillating flow fields:

$$\overline{(U^3)} = \left(\overline{U}\right)^3 = U_0{}^3$$

So we can interpret equation C.1 also in this sense:

$$\overline{(U_{OSCILLATING}(t)^3)} > \overline{(U_{NON\,OSCILLATING}(t)^3)} \tag{C.3}$$

The instantaneous power production is given by:

$$P(t) = \frac{1}{2} C_P\, \rho\, A_{rotor}\, U(t)^3$$

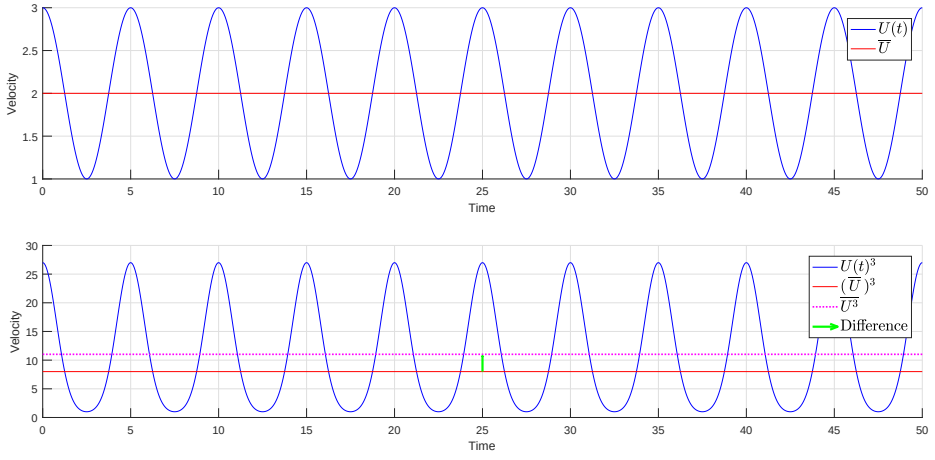


Figure C.5: Oscillating velocity (upper), average value function property (lower).

So the average power production is:

$$\overline{P(t)} = \frac{1}{2} C_P \, \rho \, A_{rotor} \, \overline{U(t)^3} \tag{C.4}$$

Substituting C.3 into C.4 it is clear that an oscillating flow field induces a higher power production than a non oscillating one.

Bibliography

[1] E. Tucker. NREL leads wind farm modeling research. *Continuum*, issue 3, 2012.

[2] R. J. Barthelmie, O. Rathmann, S. T. Frandsen, K. S. Hansen, E. Politis, J. Prospathopoulos, K. Rados, D. Cabezón, W. Schlez, and J. Phillips. Modelling and measurements of wakes in large wind farms. *Journal of Physics: Conference Series*, 75(1):012049, 2007.

[3] G. P. Corten and P. Schaak. Heat and flux: Increase of wind farm production by reduction of axial induction, 2003. Oral presentation in the session Development of Measurement Methods at the 2003 EWEC.

[4] J. Wang, C. L. Bottasso, and F. Campagnolo. Wake redirection: comparison of analytical, numerical and experimental models. *Journal of Physics: Conference Series*, 753(3): 032064, 2016.

[5] W. Munters and J. Meyers. Towards practical dynamic induction control of wind farms: analysis of optimally controlled wind-farm boundary layers and sinusoidal induction control of first-row turbines. *ind Energy Science*, 3(1):409–425, 2018. JOURNAL: Wind Energy Science.

[6] S. B. Pope. Turbulent flows. Cambridge University Press, Cambridge, 2000.

[7] M. J. Jason and L. B. J. Marshall. Fast user's guide. *NREL/EL-500e38230*, 2005.

[8] M. Churchfield, S. Lee, and P. Moriarty. Overview of the simulator for wind farm application (sowfa). 2012.

[9] P. M. O. Gebraad, F. W. Teeuwisse, J. W. van Wingerden, P. A. Fleming, S. D. Ruben, J. R. Marden, and L. Y. Pao. Wind plant power optimization through yaw control using a parametric model for wake effects—a cfd simulation study. *Wind Energy*, 19(1): 95–114, 2016.

[10] F. Campagnolo, V. Petrović, J. Schreiber, E. M. Nanos, A. Croce, and C. L. Bottasso. Wind tunnel testing of a closed- loop wake deflection controller for wind farm power maximization. *Journal of Physics: Conference Series*, 753(3):032006, 2016.

[11] E. Bossanyi. Combining induction control and wake steering for wind farm energy and fatigue loads optimisation. *Journal of Physics: Conference Series*, 1037(3), 2018.

[12] M. Steinbuch, de W. W. Boer, O. Bosgra, S. Peeters, and J. Ploeg. Optimal control of wind power plants. *Journal of Wind Engineering and Industrial Aerodynamics*, 27(1-3,): 237–246, 1988.

[13] J. Annoni, P. M. O. Gebraad, A. K. Scholbrock, P. A. Fleming, and J. van Wingerden. Analysis of axial- induction- based wind plant control using an engineering and a high-order wind plant model. *Wind Energy*, 19(6):1135–1150, 2016.

[14] P. M. O. Gebraad and J. W. van Wingerden. Maximum power-point tracking control for wind farms. *Wind Energy*, 18(3):429–447, 2015.

[15] W. Munters and J. Meyers. Dynamic strategies for yaw and induction control of wind farms based on large-eddy simulation and optimization. *Energies*, 11(1):177, 2018.

[16] W. Munters and J. Meyers. An optimal control framework for dynamic induction control of wind farms and their interaction with the atmospheric boundary layer. *Philos Trans A Math Phys Eng Sci*, 375(2091), 2017.

[17] E. A. Bossanyi. Individual blade pitch control for load reduction. *Wind Energy*, 6(2): 119–128, 2003.

[18] C. L. Bottasso, A. Croce, C. E. D. Riboldi, and M. Salvetti. Cyclic pitch control for the reduction of ultimate loads on wind turbines. *Journal of Physics: Conference Series*, 524(1):012063, 2014.

[19] D. Medici and P. H. Alfredsson. Measurements behind model wind turbines: further evidence of wake meandering. *Wind Energy*, 11(2):211–217, 2008.

[20] S. Kalvig, E. Manger, and B. Hjertager. Comparing different cfd wind turbine modelling approaches with wind tunnel measurements. *Journal of Physics: Conference Series*, 555 (1):012056, 2014.

[21] J. Jonkman, S. Butterfield, W. Musial, and G. Scott. Definition of a 5-MW reference wind turbine for offshore system development. *National Renewable Energy Laboratory, Golden, CO, Technical Report No.NREL/TP-500-38060*, 2009.

[22] Y. Yuan and J. Tang. Adaptive pitch control of wind turbine for load mitigation under structural uncertainties. *Renewable Energy*, 105:483–494, 2017.

[23] H. J. Sutherland. On the fatigue analysis of wind turbines. Technical report, Sandia National Labs., 1999.

[24] S.-E. Gryning, R. Floors, A. Pena, E. Batchvarova, and B. Bruemmer. Weibull wind-speed distribution parameters derived from a combination of wind- lidar and tall- mast measurements over land, coastal and marine sites. *Boundary-Layer Meteorology*, 159(2): 329–348, 2016.