

UNIMORE
UNIVERSITÀ DEGLI STUDI DI
MODENA E REGGIO EMILIA

Department of Engineering "Enzo Ferrari"

Master's Degree in Vehicle Engineering

Analysis of wake dynamics in floating offshore wind turbines under platform surge motion

Supervisor:
Prof. Andrea Cimarelli

Co-supervisors:
Prof. Philipp Schlatter
Dott. Riccardo Casali

Candidate:
Davide De Gennaro

Academic Year 2026/2027

Sono sempre i sogni
a dare forma al mondo

Acknowledgements

This thesis represents the culmination of a long and challenging academic journey. Reaching this point would not have been possible without the support, guidance, and opportunities provided by the people who accompanied me throughout these years.

First and foremost, I would like to express my sincere gratitude to my supervisor, Prof. Andrea Cimarelli, for his constant availability, guidance, and for giving me the opportunity to carry out this thesis abroad. I am particularly grateful for the trust he placed in me and for allowing me to undertake an experience that was valuable both academically and personally.

My sincere thanks also go to Prof. Philipp Schlatter for welcoming me to his research group and giving me the opportunity to work in such a stimulating academic environment. I am especially grateful to Riccardo Casali, who followed me throughout my work in Germany with patience, availability, and valuable scientific discussions. His support was an important part of this experience, both from an academic and personal perspective.

I would also like to thank the people I met during my time in Germany. In particular, Charlie and Daniel, with whom I had the opportunity to share many moments beyond the university, made my stay an even more enjoyable and memorable experience. I will remember not only the academic challenges of these months, but also the friendships and experiences that came with them.

A special thank you goes to my family, and especially to my parents and my sister Sara, for their unconditional support, patience, and encouragement throughout my entire academic journey. Their trust has always given me the freedom to pursue my goals, even when the path was not always straightforward. I am deeply grateful for everything they have done for me and for always being there.

I would also like to thank Sofia, for her constant support and for being by my side throughout this journey. Her presence, encouragement, and understanding have meant a great deal to me, and I am grateful to have shared these years with her.

I would also like to thank my extended family for their constant support and for sharing with me the many moments that have accompanied me throughout these years.

Finally, I would like to thank my friends. To my university friends, thank you for sharing the difficulties, the challenges, and all the lighter moments that made these years more enjoyable. And to Nico, a friend I have known since high school, thank you for always being there despite the distance and for the passion for cars that has kept us connected throughout the years.

To all of you, thank you for being part of this journey.

Abstract

Floating Offshore Wind Turbines (FOWTs) represent a promising technology for exploiting the wind resources available in deep-water regions. However, the dynamic response of the floating platform introduces additional complexity to the aerodynamic behaviour of wind turbine wakes.

This thesis investigates the wake aerodynamics of an offshore wind turbine under prescribed platform surge motions. Large Eddy Simulations (LES) are performed to reproduce the turbulent flow around the wind turbine. The Actuator Line Method (ALM) is coupled with LES to represent the aerodynamic forces exerted by the blades on the flow, allowing different turbine configurations with prescribed motion conditions to be investigated and compared. The aim is to assess the influence of platform motion dynamics on the downstream flow field and wake dynamics.

To analyse the large amount of data generated by the CFD simulations, a dedicated post-processing framework is developed. OpenFOAM simulation data are processed to perform a Hilbert Proper Orthogonal Decomposition (HPOD), providing spatial modes and their associated energy content. The resulting modes can be reconstructed in a format compatible with standard OpenFOAM files, enabling their direct visualisation in ParaView. The POD methodology is first validated on a canonical cylinder wake and the results are compared with established results from the literature.

The validated framework is subsequently applied to the wind turbine simulations, with particular attention given to the identification of the dominant coherent structures and their evolution. The results of this study can contribute to a better understanding of the relationship between floating platform motion and wake dynamics.

Contents

1 Introduction **1**

2 Numerical Modelling of Wind Turbine Aerodynamics **7**

2.1 Large-Eddy Simulation 7

2.1.1 Governing equations and spatial filtering 7

2.1.2 Subgrid-scale modelling 10

2.1.3 Smagorinsky model 11

2.2 Actuator Line Method 12

2.2.1 Motivation and positioning of ALM 12

2.2.2 Theoretical foundation of the ALM 13

2.2.3 Coupling of ALM with LES 17

3 Proper Orthogonal Decomposition **19**

3.1 Introduction 19

3.2 Proper Orthogonal Decomposition 20

3.2.1 Theoretical Background 20

3.2.2 Finite-Dimensional Formulation of POD 20

3.2.3 Singular Value Decomposition 21

3.2.4 Computing POD: Eigendecomposition, Method of Snapshots and Efficient SVD Algorithms 26

3.2.5 Weighted Inner Products and Non-Uniform Grids 28

3.2.6 Limitations of Classical POD 29

3.3 Hilbert POD 29

3.3.1 Motivation for HPOD 29

3.3.2 Hilbert Transform and Analytic Signal 30

3.3.3 HPOD formulation 31

3.3.4 Treatment of edge effects 32

3.4 Numerical implementation of the POD framework 32

3.4.1 General computational workflow 33

3.4.2 MPI data distribution and domain decomposition 33
3.4.3 Snapshots matrix assembly 35
3.4.4 Temporal processing and Hilbert transform 37
3.4.5 POD decomposition and extraction of the modes 38
3.5 Validation of HPOD on a Cylinder Wake 39
3.5.1 Cylinder Flow Configuration 39
3.5.2 HPOD Implementation and Post-Processing 41
3.5.3 HPOD Results 42

4 Wind Turbine Simulations **45**
4.1 Simulation setup 45
4.1.1 Computational domain and mesh 45
4.1.2 Turbine and actuator-line configuration 47
4.1.3 Flow and turbulent conditions 48
4.1.4 Numerical setup 49
4.1.5 Simulation cases 50
4.2 Wake characteristics of the static rotor 51
4.2.1 Wake structure and velocity field 52
4.2.2 Velocity deficit, wake recovery and meandering 53
4.3 Effect of surge motion on the wake 55
4.3.1 Comparison of static and surge cases 56
4.4 Hilbert POD analysis on wind turbine wakes 64
4.4.1 Hilbert POD analysis 65

5 Conclusions **71**
5.1 Summary of the work 71
5.2 Limitations and future work 72

List of Figures

2.1 Energy spectrum in the LES approach. Reproduced from [4]. 9
2.2 Velocity triangle and relative velocity experienced by a blade element. Reproduced from [7]. 14

3.1 Full SVD matrices shape. 22
3.2 Economy SVD matrices shape. 23
3.3 Truncated SVD matrices shape. 24
3.4 Computational domain of the cylinder test case. 40
3.5 Vortex shedding behaviour of cylinder wake. 41
3.6 First three spatial HPOD modes of the cylinder wake. The real parts are shown in the left column and the imaginary parts in the right column. 42
3.7 Components of the first HPOD mode. The top panels show the real (left) and imaginary (right) parts of the HPOD mode in the streamwise direction, while the bottom panel shows the corresponding components in the crosswise direction. 44

4.1 Computational domain of the wind turbine. 46
4.2 Discretized domain with three levels of refinement. 47
4.3 Variable blade ϵ. 48
4.4 Evolution of the rotor power during the preliminary simulation. . . . 50
4.5 Velocity field at $t = 600$ s. 52
4.6 Streamwise velocity and vorticity fields near the rotor. 53
4.7 Velocity deficit profile and Gaussian fit at $t = 500$ s and $x/D = 4$. . . 54
4.8 Velocity deficit profile and Gaussian fit at $t = 500$ s and at different downstream positions $x/D = 4, 8, 13.5, 17.5$. 55
4.9 Instantaneous (left) and mean (right) rotor power in the fixed-bottom reference case. 56
4.10 Instantaneous rotor power in the surge configuration cases (St=0.39 on the left, St=0.91 on the right). 57

4.11 Comparison of the instantaneous streamwise velocity fields for the fixed-bottom reference case and the two surge motion cases ($St = 0.39$ and $St = 0.91$). 58
4.12 Comparison of the mean streamwise velocity fields for the two surge motion cases ($St = 0.39$ and $St = 0.91$). 59
4.13 Comparison of the velocity deficit for the two surge motion cases ($St = 0.39$ and $St = 0.91$). 61
4.14 Velocity magnitude of the first POD mode: both real (top) and imaginary (bottom) parts. 65
4.15 Velocity magnitude of the second POD mode: both real (top) and imaginary (bottom) parts. 66
4.16 Streamwise (top) and transverse (bottom) velocity components of the first POD mode. 67
4.17 Magnitude (top), streamwise velocity component (centre), and transverse velocity component (bottom) of the first POD mode for the low-frequency surge configuration. 68
4.18 Magnitude (top), streamwise velocity component (centre), and transverse velocity component (bottom) of the first POD mode for the high-frequency surge configuration. 70

List of Tables

3.1 Energy content of the first five HPOD modes among the ten considered for the analysis. 43

4.1 Simulation cases considered in the present work. 51
4.2 Dominant wake meandering frequency, Strouhal number and corresponding period at different downstream positions. 55
4.3 Maximum mean streamwise velocity deficit across the cross-stream direction at different downstream positions for the fixed-bottom and surge-motion configurations. 60
4.4 Mean resolved normalised TKE, analysis at different downstream positions for the fixed-bottom and surge-motion configurations. 62
4.5 Mean absolute value of the normalized $\overline{u'v'}$ component at different downstream positions. 63
4.6 Energy content of eight HPOD modes for fixed-bottom configuration. 65
4.7 Energy content of eight HPOD modes in $St = 0.39$ configuration. . . 67
4.8 Energy content of eight HPOD modes in $St = 0.91$ configuration. . . 69

Chapter 1

Introduction

The availability of reliable and affordable energy is a fundamental requirement in the modern society. Over the past decades, the continuous growth in industrial activity and energy consumption has increased the need for energy production worldwide. At the same time, the growing awareness of the environmental impacts associated with conventional fossil-fuel-based energy systems has encouraged the development and adoption of more sustainable alternatives.

Renewable energy resources have therefore gained increasing importance in the global energy system. Among the renewable technologies available, wind energy has become one of the most mature and rapidly expanding solutions for electricity generation. Its widespread development has been supported by the large availability of wind resources, continuous technological progress, and continuously decreasing production costs.

Historically, wind power has been exploited to perform mechanical work, such as water pumping and grain milling. However, modern wind energy refers to the conversion of the kinetic energy of the wind into electrical power through wind turbines. Unlike conventional fossil-fuel power plants, where electricity generation requires the continuous consumption of raw materials, wind energy relies on a renewable resource that is naturally replenished and freely available. Consequently, the environmental impact associated with electricity production is significantly reduced while operating costs remain comparatively low after the installation.

During the last two decades, continuous technological improvements have allowed onshore wind turbines to become one of the most cost-competitive renewable energy technologies. Nevertheless, the expansion of onshore wind farms is increasingly constrained by land availability, visual impact, environmental considerations, and public acceptance. As a result, considerable attention has shifted towards offshore wind energy, where stronger and more consistent wind resources allow higher energy

production and improved capacity factors.

The first generation of offshore wind farms mainly employed fixed-bottom support structures, including monopiles, jackets, and gravity-based foundations. These solutions have proven to be technically reliable and economically competitive in shallow and intermediate waters. However, their application becomes increasingly challenging as the depth of the water increases. Beyond approximately 50–60 m, the complexity of installation, structural requirements, and foundation costs increase significantly, making fixed-bottom solutions progressively less economically attractive.

To overcome these limitations, Floating Offshore Wind Turbines (FOWTs) have emerged as a promising alternative for exploiting the wind resources available in deep-water regions. By supporting the turbine on a floating platform anchored through mooring systems (instead of rigid foundations of fixed-bottom wind turbines), FOWTs can be deployed in locations that are inaccessible to conventional offshore wind turbines. This significantly expands the geographical areas suitable for offshore wind development and offers the potential to further increase global renewable energy production while contributing to future decarbonisation goals.

Although Floating Offshore Wind Turbines (FOWTs) offer the possibility of exploiting deep-water wind resources, their dynamic behaviour is considerably more complex than that of conventional fixed-bottom offshore wind turbines. Unlike fixed structures, floating platforms are continuously subjected to environmental loads generated by wind, waves, and ocean currents. These external forces induce several platform motions, which directly influence the aerodynamic performance of the wind turbine and its interaction with the surrounding flow field.

The motion of a floating platform can be described by six degrees of freedom (6-DOF): surge, sway, heave, roll, pitch, and yaw. The magnitude of these motions depends on several factors, including the platform type, the mooring system, sea state, and atmospheric conditions. Among these motions, surge and pitch are generally recognised as the most influential on the aerodynamic behaviour of the rotor, as they modify the relative velocity and orientation of the incoming wind experienced by the blades [1].

The aerodynamic performance of a wind turbine is governed by the interaction between the incoming atmospheric flow and the rotating blades. As the wind passes through the rotor, part of its kinetic energy is extracted and converted into mechanical power, resulting in a reduction of the wind velocity downstream of the turbine. This region, commonly referred to as the wake, is characterized by a velocity deficit, increased turbulence intensity, and the presence of coherent vortical

structures generated by the rotating blades.

Compared with fixed-bottom offshore wind turbines, the wake generated by a Floating Offshore Wind Turbine is considerably more complex. Indeed, platform motions may accelerate or delay wake recovery, modify the velocity deficit, enhance wake meandering, and influence the formation and evolution of large-scale vortical structures, leading to fluctuations in aerodynamic loads, power production, and thrust [1]. As a result, the wake generated by a floating turbine is highly dependent on both the characteristics of the incoming atmospheric flow and the dynamic response of the floating platform. Therefore, accurate prediction of the aerodynamic behaviour of FOWTs requires consideration of the interaction between the aerodynamic, hydrodynamic, structural, and control-system responses.

Previous studies have shown that wake interactions can significantly decrease the energy production of downstream turbines while simultaneously increasing fatigue loads on critical structural components [2]. In particular, the modification of the turbine wake due to platform motion has attracted increasing attention, since it directly affects not only the performance of an individual turbine but also the behaviour and efficiency of entire offshore wind farms. Understanding these wake dynamics requires an accurate characterization of the turbulent flow structures developing downstream of the rotor. Consequently, a variety of experimental and numerical approaches have been developed to investigate the complex aerodynamics of floating offshore wind turbines.

The present work investigates the aerodynamic behaviour of a Floating Offshore Wind Turbine by analysing the influence of prescribed platform motions on the evolution of its wake. The analysis focuses on surge motion, which, among the six platform degrees of freedom, has been identified as the dominant degree of freedom affecting rotor aerodynamics and wake dynamics. The surge motion changes the relative wind velocity encountered by the rotor, generating periodic variations in the aerodynamic load of the blades and may significantly influence the structure and stability of the downstream wake. However, the literature does not yet provide a unanimous conclusion regarding its overall impact: while some numerical and experimental studies report an accelerated wake recovery associated with enhanced turbulent mixing, others observe a persistent velocity deficit and reduced wake recovery depending on the operating conditions, motion characteristics, and modelling approach adopted. This lack of consensus highlights the need for further investigation into the aerodynamic behaviour of floating wind turbines under realistic operating conditions [2].

Over the last decade, a wide range of numerical and experimental techniques have

been developed to investigate the aerodynamics and wake characteristics of FOWTs. Experimental studies have been conducted using scaled wind turbine models tested in wind tunnels, wave basins, and hybrid experimental facilities. These experiments have provided valuable insight into the interaction between platform motions, aerodynamic loading, and wake development, while also supplying high-quality datasets for the validation of numerical models. However, reproducing realistic offshore operating conditions at laboratory scale remains challenging due to scaling effects, the simultaneous representation of wind and wave loading, and the complexity of accurately modelling atmospheric turbulence. To complement these investigations, several numerical approaches have been developed: for example, engineering methods based on Blade Element Momentum (BEM) theory remain widely used because of their computational efficiency and suitability for aeroelastic simulations. Nevertheless, their simplified representation of the flow field limits their capability to accurately capture highly unsteady wake dynamics. Among the available numerical approaches, high-fidelity Computational Fluid Dynamics (CFD) simulations, particularly those employing Large Eddy Simulation (LES), provide a more detailed description of the turbulent flow by directly resolving the largest energy-containing eddies while modelling only the smallest turbulent scales, making it particularly suitable for investigating the effects of platform motion on the downstream flow field. Within this framework, the Actuator Line Method (ALM) has become one of the most widely adopted approaches for representing rotating wind turbine blades without explicitly resolving their geometry, allowing accurate wake prediction at a computational cost that remains feasible for practical applications.

However, the vast amount of data generated by LES poses new challenges for the analysis and interpretation of complex wake dynamics. Therefore, reduced-order techniques, such as Proper Orthogonal Decomposition (POD) and its extension based on the Hilbert transform (HPOD), provide an effective framework for identifying the dominant coherent structures governing wake evolution and for analysing the influence of platform motion on the aerodynamic behaviour of floating offshore wind turbines.

The main objective of this thesis is to compare the wake characteristics of floating and fixed offshore wind turbines to evaluate how surge motion modifies the wake development. Particular attention is devoted to the evolution of the velocity field and the dominant coherent structures within the wake, providing a better understanding of the mechanisms responsible for the differences observed between floating and fixed configurations.

The scope of this work is limited to the aerodynamic analysis of the wake gener-

ated by an isolated floating offshore wind turbine. The turbine is modelled without explicitly including the tower and floating platform, while the surge motion is prescribed as known inputs. This approach enables the isolated investigation of the influence of platform motions on the surrounding flow field without considering the hydrodynamic behaviour of the platform or the structural response of the turbine.

The thesis is organised into five chapters, each addressing a specific aspect of the aerodynamic analysis of Floating Offshore Wind Turbines.

Chapter 2 describes the simulation methodology adopted in this work. The numerical framework used for the simulations is introduced, including the Large Eddy Simulation approach and the Actuator Line Method for turbine representation.

Chapter 3 describes the post-processing methodology based on the Hilbert Proper Orthogonal Decomposition, describing the theory behind this method, addressing the numerical framework and showing an application example on a cylinder wake.

Chapter 4 presents the numerical results obtained from the simulations of Floating Offshore Wind Turbines. The wake characteristics of the floating and fixed turbine configurations are compared, with particular attention given to the effects of surge motion on velocity fields, wake recovery and turbulence characteristics. HPOD is then employed to further investigate the flow characteristics, and the results are interpreted in relation to the two previous investigations.

Finally, Chapter 5 summarizes the main conclusions of the thesis and provides suggestions for future developments, including possible extensions of the numerical methodology and further investigations of floating offshore wind turbine wake dynamics.

Chapter 2

Numerical Modelling of Wind Turbine Aerodynamics

2.1 Large-Eddy Simulation

Turbulent flows can be simulated using different approaches, depending on the level of detail required and the available computational resources. Among these approaches, Large-Eddy Simulation (LES) provides a suitable compromise between the level of detail and the computational cost for the analysis of complex turbulent flows.

Large-Eddy Simulation is a computational approach used in fluid dynamics simulations for modelling turbulent flows. By using this approach, large and energy-containing structures are explicitly resolved, while the effects of the smaller and unresolved scales are modelled. Compared with Reynolds-Averaged Navier–Stokes (RANS) approaches, LES provides a more detailed description of the unsteady turbulent flow, while requiring significantly lower computational resources than Direct Numerical Simulation (DNS). This makes LES particularly suitable for the analysis of complex and highly unsteady flows, such as wind-turbine wakes.

2.1.1 Governing equations and spatial filtering

The governing equations for an incompressible Newtonian fluid can be written in the following form:

$$\begin{cases} \dfrac{\partial u_i}{\partial x_i} = 0 \\ \dfrac{\partial u_i}{\partial t} + \dfrac{\partial (u_i u_j)}{\partial x_j} = -\dfrac{1}{\rho}\dfrac{\partial p}{\partial x_i} + \nu \dfrac{\partial^2 u_i}{\partial x_j \partial x_j} \end{cases} \tag{2.1}$$

where u_i and p denote the velocity components and pressure, respectively. In turbulent flows, the velocity field contains a wide range of spatial and temporal scales. While Direct Numerical Simulations (DNS) aim to resolve all these scales, Large-Eddy Simulation (LES) separates the flow into large-scale structures, which are explicitly resolved, and smaller scales, whose effects are modelled by using a subgrid-scale model.

The resolved field is obtained by applying a spatial filtering operation to the instantaneous flow field. This operation can be expressed in the form of a *Convolution Integral*:

$$\bar{u}_i(x_j, t) = \int G(r_j, x_j)\, u_i(x_j - r_j, t)\, dr_j \tag{2.2}$$

where G is the filter kernel, x_j denotes the location at which the filtered velocity is evaluated, and r_j represents the spatial distance from this location. Consequently, the filtered velocity at x_j depends on the velocity field in the surrounding region, with each contribution weighted according to the filtering kernel, and the extent over which the surrounding flow field contributes to the filtered velocity is related to the characteristic filter width Δ.

The filtering operation provides a *decomposition* of the instantaneous velocity field into a large-scale resolved component and a subgrid-scale contribution:

$$u_i(x_j, t) = \bar{u}_i(x_j, t) + u_i^{SGS}(x_j, t) \tag{2.3}$$

The spatial filtering introduced above should be distinguished from the statistical decomposition of the velocity field into the mean and fluctuating components. For the analysis of the flow fields, the instantaneous velocity can be expressed as:

$$u_i(\boldsymbol{x}, t) = U_i(\boldsymbol{x}) + u_i'(\boldsymbol{x}, t) \tag{2.4}$$

where U_i denotes the mean velocity component and u_i represents the corresponding velocity fluctuation.

Different filtering kernels can be adopted, such as the top-hat, Gaussian or cut-off filters [3]. Although these filters differ on the spatial weighting, their common purpose is to separate the large-scale resolved structures from the smaller-scales whose effects require modelling.

Applying the filtering operation to the Navier–Stokes equations, it is possible to retrieve the so-called *filtered Navier–Stokes equations*:

$$\begin{cases} \dfrac{\partial \bar{u}_i}{\partial x_i} = 0 \\ \dfrac{\partial \bar{u}_i}{\partial t} + \dfrac{\partial (\bar{u}_i \bar{u}_j)}{\partial x_j} = -\dfrac{1}{\rho}\dfrac{\partial \bar{p}}{\partial x_i} + \nu \dfrac{\partial^2 \bar{u}_i}{\partial x_j \partial x_j} - \dfrac{\partial \tau_{ij}}{\partial x_j} \end{cases} \tag{2.5}$$

where $\bar{u}_i$ and $\bar{p}$ denote the filtered velocity and pressure, respectively. The additional term τ_{ij} is the *subgrid-scale stress tensor*, defined as follows:

$$\tau_{ij} = \overline{u_i\, u_j} \; - \; \bar{u}_i \bar{u}_j \tag{2.6}$$

This term arises from the filtering of the nonlinear convective term and represents the influence of the unresolved scales on the resolved flow. Since the quantity $\overline{u_i\, u_j}$ cannot be directly obtained from the resolved velocity field, the SGS stress tensor must be modelled.

The physical interpretation of the separation between resolved and unresolved scales can be further illustrated in the wavenumber domain through the *turbulent kinetic energy spectrum*. The quantity $E(k)$ represents the distribution of turbulent kinetic energy among different wavenumbers. The total turbulent kinetic energy is obtained by integrating the energy spectrum over all wavenumbers:

$$K = \int_0^\infty E(k)\, dk \tag{2.7}$$

where the wavenumber k is related to the characteristic length scale λ of a turbulent structure by:

$$k = \frac{2\pi}{\lambda} \tag{2.8}$$

Consequently, low wavenumbers correspond to large turbulent structures, while high wavenumbers represent progressively smaller structures.

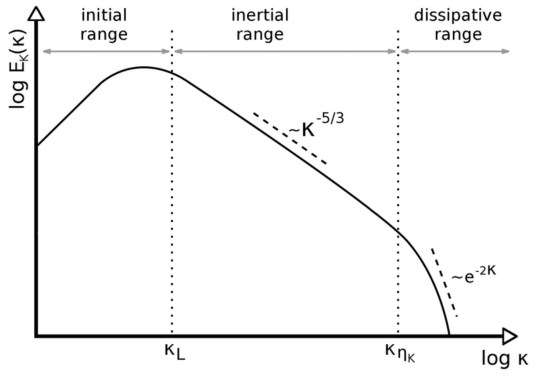


Figure 2.1: Energy spectrum in the LES approach. Reproduced from [4].

In LES, the filter introduces a separation between the large scales that are resolved by the computational grid and the smaller scales whose effects are represented through the SGS model. The corresponding energy distribution can therefore be represented schematically as shown in Figure 2.1.

2.1.2 Subgrid-scale modelling

As discussed in the previous section, the filtering operation introduces the subgrid-scale tensor τ_{ij}, which accounts for the effects of the unresolved scales on the resolved flow. Since these effects cannot be directly determined from the resolved velocity field, a suitable modelling approach is required.

From an energetic point of view, the main role of the SGS model is to account for the transfer and dissipation of the turbulent kinetic energy associated with the unresolved scales. In general, the SGS stress tensor can be expressed as a function of the resolved flow field, the filter width and the filter kernel:

$$\tau_{ij} = \tau_{ij}(\bar{u}_i, \bar{p}, \Delta, G) \tag{2.9}$$

The dissipative effect of the unresolved scales can be modelled by introducing an eddy-viscosity approach. Therefore, a possible way to model the deviatoric part of the subgrid-scale (SGS) stress tensor is:

$$\tau_{ij} - \frac{1}{3}\tau_{kk}\delta_{ij} = -2\nu_{SGS}\bar{S}_{ij} \tag{2.10}$$

where τ_{kk} is the trace of the SGS stress tensor, δ_{ij} is the Kronecker delta, ν_{SGS} is the so-called *subgrid-scale eddy viscosity*, and $\bar{S}_{ij}$ is the filtered strain-rate tensor, defined as:

$$\bar{S}_{ij} = \frac{1}{2}\left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i}\right) \tag{2.11}$$

By introducing this modelling approach into the filtered Navier–Stokes equations, the momentum equation can be written in the form:

$$\frac{\partial \bar{u}_i}{\partial t} + \frac{\partial(\bar{u}_i\bar{u}_j)}{\partial x_j} = -\frac{1}{\rho}\frac{\partial \bar{p}}{\partial x_i} + \frac{\partial}{\partial x_j}\left[2(\nu + \nu_{SGS})\bar{S}_{ij}\right] \tag{2.12}$$

Therefore, the SGS viscosity acts as an additional diffusion mechanism that accounts for the fluctuations occurring at small scales.

2.1.3 Smagorinsky model

The Smagorinsky model is an eddy-viscosity model in which the SGS viscosity is related to a characteristic mixing length and a characteristic velocity scale. Therefore, the SGS viscosity can be expressed as:

$$\nu_{SGS} = f(l, v) \tag{2.13}$$

where l is the characteristic mixing length and v is the corresponding velocity scale. The SGS viscosity is a local turbulent quantity and therefore varies in both space and time. It can be estimated as the square of a characteristic mixing length divided by a mixing time:

$$\nu_{SGS} = l\, v = l\, \frac{l}{\tau} = \frac{l^2}{\tau} \tag{2.14}$$

Assuming that $\frac{1}{\tau} \sim |\bar{S}_{ij}|$, it is possible to retrieve the following relation:

$$\nu_{SGS} = l^2 |\bar{S}_{ij}| \tag{2.15}$$

where $|\bar{S}_{ij}| = \sqrt{2\bar{S}_{ij}\bar{S}_{ij}}$ and l is the mixing length. The mixing length is assumed to be proportional to the filter width:

$$l = C_s \Delta \tag{2.16}$$

where by choosing the value of the filter length Δ it is possible to impose which scales are resolved and which scales are modelled, while C_s is the *Smagorinsky model constant*. Substituting this relation into the expression for the SGS viscosity, it is possible to obtain:

$$\begin{aligned} \tau_{ij} - \frac{1}{3}\tau_{kk}\delta_{ij} &= -2\nu_{SGS}\bar{S}_{ij} \\ &= -2(C_s\Delta)^2\, |\bar{S}_{ij}|\bar{S}_{ij} \end{aligned} \tag{2.17}$$

However, the formulation implemented in the present work differs slightly from the classical Smagorinsky model. Instead of obtaining the characteristic velocity scale directly from the local strain rate, the formulation used in OpenFOAM derives it from the subgrid-scale turbulent kinetic energy, k_{SGS}. The SGS viscosity is therefore modelled as:

$$\nu_{SGS} = C_k \Delta \sqrt{k_{SGS}} \tag{2.18}$$

where C_k is a model coefficient and $\sqrt{k_{SGS}}$ provides a characteristic velocity scale for the unresolved turbulent motion. The SGS kinetic energy is obtained from an algebraic equilibrium assumption between its production and dissipation, without introducing an additional transport equation. Further details on the implementation are provided in the OpenFOAM documentation [5].

2.2 Actuator Line Method

2.2.1 Motivation and positioning of ALM

Wind turbine loads and performance calculations have long been dominated by the Blade Element Momentum theory (BEM), first formulated by Glauert [6]. The method divides the flow into annular control volumes, applies momentum balance and energy conservation independently in each control volume (in each annulus), and computes aerodynamic forces from tabulated airfoil data (derived from wind tunnel measurements and corrected for three-dimensional effects). Although this method is simple and has a low computational cost, it has limitations in representing the various flow situations encountered in practice. Therefore, different empirical corrections have been introduced to account for phenomena such as dynamic inflow, yaw misalignment, tip loss, and heavily loaded rotors.

To address these weaknesses, more advanced inviscid approaches have been developed. One of them is the vortex wake method, in which the shed vorticity in the wake is used to compute the induced velocity field. This approach offers a more physically consistent representation of wake dynamics compared to BEM. However, free-wake formulations are more computationally expensive and prone to numerical divergence due to the intrinsic singularities of vortex filaments.

Thus, more advanced approaches have been developed, and one of them is the Actuator Disk Method (ADM). The ADM represents a more direct extension of the BEM model: it retains the use of tabulated airfoil data but replaces the annular momentum balance with the solution of a full set of Euler or Navier–Stokes equations, yielding improved accuracy. However, its fundamental limitation is that the blade forces are smeared uniformly over the disk, losing all the information about the individual blade geometry and the 3D structure of the tip vortices.

To overcome this limitation, Sørensen and Shen (2002) introduced the Actuator Line Method (ALM), which combines a three-dimensional Navier–Stokes solver with a body force distribution applied along the lines representing the individual rotor blades. Rather than resolving the blade geometry on the computational grid, each blade is replaced by a radial line along which aerodynamic forces, computed from

local angles of attack and tabulated airfoil polars, are projected back onto the flow field as momentum source terms. The wake kinematics is therefore determined by a full three-dimensional simulation, while the viscous blade effects are introduced only as integrated quantities through the airfoil data. This formulation preserves the computational tractability of the actuator disk approach while enabling the resolution of individual blade wakes, tip vortex dynamics, and the three-dimensional structure of the rotor near-field, all of which are essential for the accurate aerodynamic modelling of FOWTs operating under platform motion, turbulent atmospheric inflow, and wind farm interaction.

2.2.2 Theoretical foundation of the ALM

The fundamental idea behind the ALM is that the blade geometry is not explicitly represented in the computational mesh. Instead, each blade is replaced by a line, where aerodynamic forces are calculated using the blade element theory. These forces are then introduced into the flow equations as body forces.

Since the blade geometry is not explicitly resolved, the ALM does not capture the detailed flow structures associated with the surface, such as the boundary layer, but instead, aerodynamic loadings are represented through the distributed forces.

The ALM formulation presented by Sørensen and Shen couples the actuator line representation of the rotor blades with three-dimensional Navier–Stokes equations. In the original formulation, the governing equations are formulated in the vorticity-velocity formulation ($\boldsymbol{\omega} - \boldsymbol{u}$) and solved in a cylindrical coordinate system (r, θ, z). The formulation is obtained by applying the curl operator to the Navier–Stokes equations in primitive variables, yielding a set of three vorticity transport equations (2.19), three Cauchy-Riemann equations linking velocity and vorticity (2.20), and a continuity equation (2.21).

$$\frac{\partial \boldsymbol{\omega}}{\partial t} + \nabla \times (\boldsymbol{\omega} \times \boldsymbol{u}) = -\nu \nabla \times (\nabla \times \boldsymbol{\omega}) + \nabla \times \boldsymbol{f_\epsilon} \tag{2.19}$$

$$\nabla \times \boldsymbol{u} = \boldsymbol{\omega} \tag{2.20}$$

$$\nabla \cdot \boldsymbol{u} = 0 \tag{2.21}$$

where ν is the kinematic viscosity and $\boldsymbol{f_\epsilon}$ is the load introduced as a body force on the right hand side of the vorticity transport equations.

Unlike the filtered Navier–Stokes formulation introduced in Section 2.1, the key advantage of this formulation is that the pressure disappears entirely from the system: indeed, the curl of a gradient is identically zero ($\nabla \times \nabla p = 0$). Therefore,

applying the curl operator to the momentum equations eliminates the pressure term; consequently the continuity equation is automatically satisfied since any field of the form $\nabla \times \boldsymbol{A}$ is divergence-free by construction [1].

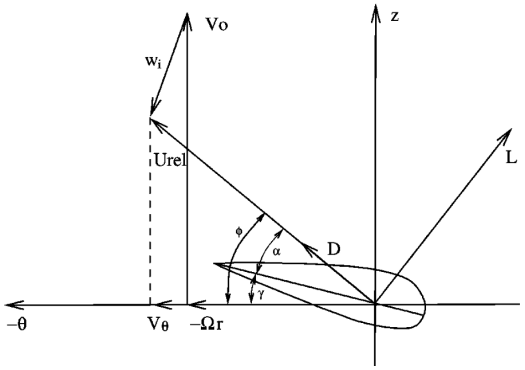


Figure 2.2: Velocity triangle and relative velocity experienced by a blade element. Reproduced from [7].

Aerodynamic forces are computed using a blade-element approach combined with two-dimensional airfoil data. At each actuator point, the local fluid velocity is interpolated from the CFD solution and combined with the blade rotational velocity to obtain the relative velocity seen by the blade element. In the inertial frame, this relative velocity can be expressed in terms of its local tangential (u_θ) and axial (u_z) flow components as:

$$\boldsymbol{U}_{rel} = (V_\theta - \Omega r, V_z) \tag{2.22}$$

where $\boldsymbol{U}_{rel}$ is the relative velocity experienced by the rotating blades, Ω is the rotor angular velocity (rad/s) and r is the radial distance of the blade element considered from the hub centre.

The local angle of attack (α), as shown in Figure 2.2, is then obtained from the relative flow angle (ϕ) and the local pitch angle (γ):

$$\alpha = \phi - \gamma \tag{2.23}$$

The force per unit span is evaluated using two-dimensional airfoil aerodynamic data, expressed as functions of the local angle of attack and Reynolds number, resulting in the lift and drag contributions:

[1] The vorticity-velocity formulation relies on two fundamental identities: $\nabla \times (\nabla p) = 0$ (the curl of a gradient is identically zero), which eliminates the pressure term when applying the curl to the momentum equation, removing the pressure-velocity coupling problem, and $\nabla \cdot (\nabla \times \mathbf{A}) = 0$ (the divergence of a curl is identically zero), which ensures the divergence-free condition when reconstructing the velocity field from the vorticity.

$$\boldsymbol{f_{2D}} = \frac{d\boldsymbol{F}}{dr} = \frac{1}{2}\rho\, u_{rel}^2\, c(C_L \boldsymbol{e}_L + C_D \boldsymbol{e}_D) \tag{2.24}$$

where $u_{rel} = |\boldsymbol{u}_{rel}|$, $C_L = C_L(\alpha, Re)$ and $C_D = C_D(\alpha, Re)$ are the lift and the drag coefficients respectively, c is the chord length, and $\boldsymbol{e}_L$ and $\boldsymbol{e}_D$ denote the unit vectors in the lift and drag directions.

The lift and drag coefficients are determined from measured or computed two-dimensional airfoil data as functions of the local angle of attack and Reynolds number. Since these data are typically derived for two-dimensional airfoil analysis or experiments, suitable corrections may be applied to account for three-dimensional rotational effects and finite aspect ratio effects, thereby providing a more realistic representation of the aerodynamic loads on rotating blades. In particular, rotational effects tend to delay flow separation, increasing lift compared with two-dimensional airfoil data, while finite aspect ratio corrections account for the influence of the finite blade span on the aerodynamic coefficients.

Considering a differential volume in cylindrical coordinates, $dv = rdrd\theta dz$, the aerodynamic force per unit volume is obtained as:

$$\boldsymbol{f} = \frac{d\boldsymbol{F}}{dv} = \frac{\boldsymbol{f}_{2D}}{r\, d\theta\, dz} \tag{2.25}$$

The resulting force is projected onto the cylindrical coordinate directions, giving the corresponding source terms:

$$f_r = 0 \tag{2.26}$$

$$f_\theta = \frac{\rho\, c\, u_{rel}^2}{2\, r\, d\theta\, dz}(C_L \sin\phi - C_D \cos\phi) \tag{2.27}$$

$$f_z = \frac{\rho\, c\, u_{rel}^2}{2\, r\, d\theta\, dz}(C_L \cos\phi + C_D \sin\phi) \tag{2.28}$$

These components define the aerodynamic body forces per unit volume that are introduced into the governing flow equations solved by the CFD model.

A critical aspect of the method is how these discrete blade forces are introduced into the continuous flow field. The aerodynamic forces computed at discrete points along the actuator line cannot be applied directly to the flow field as point forces. In the vorticity-velocity formulation (2.19), the body force term appears on the right-hand side of the vorticity transport equation as a source term through its curl: applying a point force would act as a singular vorticity source, and on a finite computational grid, this may produce severe numerical instabilities.

To avoid this singularity, the discrete force f_ϵ is found by taking the convolution of the computed normal load f and a Gaussian regularization kernel η_ϵ, which smoothly distributes the loading over a finite volume around each force point.

$$\boldsymbol{f_\epsilon} = \boldsymbol{f} \otimes \eta_\epsilon \tag{2.29}$$

where:

$$\eta_\epsilon(d) = \frac{1}{\epsilon^3 \pi^{3/2}} \exp[-(d/\epsilon)^2] \tag{2.30}$$

and where d denotes the distance between a point in the flow field and the force point on the actuator line. Therefore, the convolution (2.29) replaces each discrete actuator force point with a smooth three-dimensional Gaussian force distribution, preserving the total applied force and distributing the load over a finite volume (rather than concentrating it at a single point).

The parameter ϵ, known as the smoothing length, defines the width of the Gaussian kernel used to distribute the aerodynamic forces onto the computational grid and its choice represents a fundamental trade-off in ALM simulations. If ϵ is too small with respect to the local grid spacing, the Gaussian is under-represented, leading to numerical oscillations and causing numerical noise. Conversely, excessively large values of ϵ spread the blade loading over an unrealistic region, reducing the accuracy of the induced velocity field in the rotor plane and, in the limit case, degrading ALM towards an actuator disk [8].

Since the blade loading is distributed over a characteristic length comparable to the local chord, ϵ should ideally scale with the local chord length c. Several studies have proposed the criterion $\epsilon/c \approx 0.14 - 0.25$ as a reasonable compromise between numerical stability and physical accuracy [9].

The choice of ϵ is particularly important near the blade root and tip, where the Gaussian kernel may extend beyond the physical blade and modify the local force distribution. Near the tip, this can alter the local loading distribution and consequently the strength of the tip vortices. Therefore, specific tip loss corrections may be adopted in some ALM implementations to account for those effects and improve the prediction of aerodynamic loads. In the present work, no additional tip loss correction is applied, as the grid resolution in the blade tip region is sufficiently fine to limit these effects.

2.2.3 Coupling of ALM with LES

In the present FOWT simulations, the ALM is coupled with Large-Eddy Simulation (LES). The filtered incompressible Navier–Stokes equations, which govern the resolved velocity field $\bar{\boldsymbol{u}}$, take the form:

$$\nabla \cdot \bar{\boldsymbol{u}} = 0 \tag{2.31}$$

$$\frac{\partial \bar{\boldsymbol{u}}}{\partial t} + (\bar{\boldsymbol{u}} \cdot \nabla)\bar{\boldsymbol{u}} = -\frac{1}{\rho}\nabla \bar{p} + \nu \nabla^2 \bar{\boldsymbol{u}} - \nabla \cdot \boldsymbol{\tau} + \boldsymbol{f}_\epsilon \tag{2.32}$$

where $\boldsymbol{\tau}$ is the sub-grid scale stress tensor and $\boldsymbol{f}_\epsilon$ is the regularized body force introduced by the ALM, as defined in (2.29).

The coupling between ALM and LES is performed through an iterative loop that exchanges information between the actuator line and the resolved flow field. At each time step, the resolved local velocity field $\bar{\boldsymbol{u}}$ is sampled from the LES grid at each actuator line point. This velocity is then combined with the local blade motion to determine the relative velocity and the relative angle of attack seen from each blade element. The aerodynamic forces (lift and drag) are then obtained from prescribed airfoil polars and then distributed onto the surrounding cells via the Gaussian kernel described in Section 2.2.2. These forces enter the LES momentum equations through the body-force term $\boldsymbol{f}_\epsilon$ and modify the resolved flow field that will be used for the next iteration.

For the surge-motion cases considered in the present work, the actuator-line points are subjected to an imposed platform motion and displaced accordingly: the resulting motion of the actuator lines is therefore accounted when evaluating the local relative velocity.

Chapter 3

Proper Orthogonal Decomposition

3.1 Introduction

Modern CFD simulations provide a high level of detail in the description of complex flow phenomena, but they also introduce significant challenges related to the management, analysis, and physical interpretation of the large amount of data generated. This issue is particularly relevant in turbulent flows, where many flow structures interact over different spatial and temporal scales, leading to highly nonlinear flow behaviour.

As a consequence, an important goal in fluid dynamics is to develop methods that can simplify these complex flow fields while preserving their most important physical features. These methods reduce the complexity of the flow by representing it with a limited number of dominant structures, making it easier to analyse and understand the main features of the flow. They are also widely used for reduced-order modelling, flow control, and optimization. Their applications include flow field reconstruction and the development of reduced-order models that reproduce the main flow dynamics with a much lower computational cost than full-order CFD simulations.

Among these approaches, data-driven decomposition techniques have become widely used because they can extract useful information directly from numerical or experimental data, without requiring prior knowledge of the flow structures. One of the most widely used techniques is Proper Orthogonal Decomposition (POD) [10].

In the analysis of floating offshore wind turbine (FOWT) wakes, identifying the dominant coherent structures is essential to better understand the complex dynamics generated by the interaction between the wind, the moving rotor, and the platform motion. However, classical POD has some limitations when dealing with travelling structures, which are common in wake flows. For this reason, the Hilbert Proper

Orthogonal Decomposition (HPOD) [11] is used in this work. By including phase information through the Hilbert transform, HPOD can better describe propagating flow structures. It is therefore used to analyse the wake dynamics of the simulated FOWTs' wake.

3.2 Proper Orthogonal Decomposition

3.2.1 Theoretical Background

Given a dataset that depends on both space and time, POD provides a mathematically rigorous way to identify the dominant spatial structures, or *modes*, that contain most of the flow energy.

The fundamental idea underlying POD is a *decomposition*: a given field quantity $\boldsymbol{y}(x,t)$, depending on both the spatial position x and time t, is represented as:

$$\boldsymbol{y}(x,t) = \sum_{j=1}^{m} \boldsymbol{\phi}_j(x)\, a_j(t) \tag{3.1}$$

where $\boldsymbol{\phi}_j(x)$ are purely spatial functions, referred to as *POD modes*, and $a_j(t)$ are the corresponding *temporal coefficients*.

The objective of POD is to identify a set of modes that contains the maximum possible amount of information about the original data. The modes are ordered according to their contribution to the total energy of the system, with the first modes representing the dominant structures of the flow. By retaining only the most energetic modes, the original dataset can therefore be approximated using a reduced number of modes.

To introduce the mathematical formulation of POD, it is convenient to first consider the finite-dimensional representation of the problem.

3.2.2 Finite-Dimensional Formulation of POD

Although POD is originally introduced in a continuous framework [10], numerical CFD simulations provide the flow field only at discrete spatial locations (grid points) and at discrete time instants. Therefore, the continuous formulation must be translated into a finite-dimensional representation suitable for computational analysis. This representation is based on a collection of flow fields at different instants, commonly referred to as *snapshots*.

Suppose measurements (or simulation outputs) of a scalar field $y(x,t)$ are available at n discrete spatial locations $x_1, x_2, \ldots, x_n$ and at m discrete time instants

$t_1, t_2, \ldots, t_m$, these data are assembled into a *snapshot matrix* $Y \in \mathbb{R}^{n\times m}$, that can be represented as follows:

$$Y = \begin{bmatrix} y(x_1,t_1) & y(x_1,t_2) & \cdots & y(x_1,t_m) \\ y(x_2,t_1) & y(x_2,t_2) & \cdots & y(x_2,t_m) \\ \vdots & \vdots & \ddots & \vdots \\ y(x_n,t_1) & y(x_n,t_2) & \cdots & y(x_n,t_m) \end{bmatrix}$$

where each column represents the spatial field at a given time instant, a single *snapshot*, and each row tracks the temporal evolution at a fixed spatial location.

This construction is generalisable to vector fields. In the case of a three-dimensional velocity field, for instance, all the components $u(\boldsymbol{x},t)$, $v(\boldsymbol{x},t)$ and $w(\boldsymbol{x},t)$ are stacked into a single column, producing an augmented snapshot matrix $Y \in \mathbb{R}^{3n\times m}$:

$$Y = \begin{bmatrix} u(x_1,t_1) & u(x_1,t_2) & \cdots & u(x_1,t_m) \\ \vdots & \vdots & \ddots & \vdots \\ u(x_n,t_1) & u(x_n,t_2) & \cdots & u(x_n,t_m) \\ v(x_1,t_1) & v(x_1,t_2) & \cdots & v(x_1,t_m) \\ \vdots & \vdots & \ddots & \vdots \\ v(x_n,t_1) & v(x_n,t_2) & \cdots & v(x_n,t_m) \\ w(x_1,t_1) & w(x_1,t_2) & \cdots & w(x_1,t_m) \\ \vdots & \vdots & \ddots & \vdots \\ w(x_n,t_1) & w(x_n,t_2) & \cdots & w(x_n,t_m) \end{bmatrix}$$

This stacking procedure ensures that each component is treated simultaneously within a single decomposition, so that the resulting POD modes capture coherent spatial structures across all field components at once.

3.2.3 Singular Value Decomposition

The decomposition of the snapshot matrix Y into spatial modes and temporal coefficients is achieved through the *Singular Value Decomposition* (SVD). Before introducing the truncated form that is directly employed in POD, it is instructive to build up the decomposition progressively, starting from its most general formulation.

Full SVD

Any matrix $Y \in \mathbb{R}^{n\times m}$ admits a factorisation of the form

$$Y = U\Sigma V^* \tag{3.2}$$

regardless of its dimensions, rank, or whether it is square or rectangular. This generality makes SVD particularly suitable for the analysis of data matrices, such as the snapshot matrix Y used in POD.

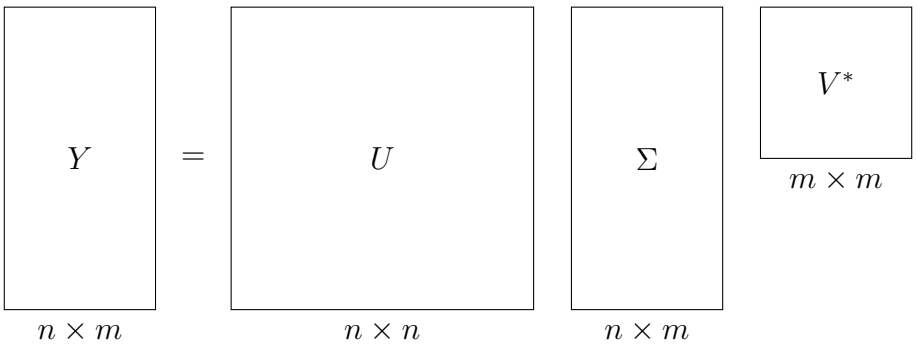


Figure 3.1: Full SVD matrices shape.

The matrices involved in the decomposition are:

- $U \in \mathbb{R}^{n \times n}$ is a unitary[2] matrix whose columns $\boldsymbol{u}_1, \boldsymbol{u}_2, \ldots, \boldsymbol{u}_n$ are called the *left singular vectors*;
- $\Sigma \in \mathbb{R}^{n \times m}$ is a rectangular diagonal matrix whose non-zero entries $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_p \geq 0$, with $p = \min(n, m)$, are called *singular values*;
- $V^* \in \mathbb{R}^{m \times m}$ is a unitary matrix whose rows are the conjugate transposes of the *right singular vectors* $\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_m$.

The SVD also admits a useful geometrical interpretation: the matrices U and V^* perform rotations of the data space without any distortion, while Σ scales the different directions according to the singular values. Therefore, the magnitude of each contribution in the decomposition is determined by the corresponding singular value[3].

The singular values σ_j are ordered in non-increasing fashion by convention:

$$\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_p \geq 0. \tag{3.3}$$

[2]A matrix Q is said to be *unitary* if $Q^*Q = QQ^* = I$, meaning that its columns (and rows) form an orthonormal basis. Geometrically, multiplication by a unitary matrix is a *pure rotation* of the space: it preserves lengths and angles.

[3]The SVD can also be interpreted geometrically: (1) V^* rotates the input space; (2) Σ stretches (or compresses) along each coordinate axis, with stretching factors $\sigma_1 >= \sigma_2 >= ...$; (3) U rotates the result into the output space. The singular values therefore measure how strongly Y acts along each of these directions. A large σ_j means Y has a strong "action" along the j-th mode; a small σ_j means that direction contributes very little to Y.

This ordering is not arbitrary. The singular values are uniquely determined by the matrix Y: they are the square roots of the eigenvalues of the symmetric positive semi-definite matrix Y^*Y (or equivalently YY^*), which are guaranteed to be real and non-negative. Since eigenvalues can always be sorted, the singular values are arranged from largest to smallest, so that the most energetically significant contributions appear first. The precise meaning of "energetically significant" will become clear in Section 3.2.3.

Economy SVD

As stated above, the full SVD of $Y \in \mathbb{R}^{n\times m}$ involves a square matrix $U \in \mathbb{R}^{n\times n}$ and a rectangular diagonal matrix $\Sigma \in \mathbb{R}^{n\times m}$. However, in fluid mechanics applications, the number of spatial degrees of freedom n is usually much larger than the number of available snapshots m [4], i.e. $n \gg m$. Therefore, Σ can contain at most $p = \min(n, m) = m$ non-zero singular values, meaning that only the first m columns of U can contribute to the decomposition. The remaining columns are associated with singular zero values and therefore do not affect the product $U\Sigma V^*$.

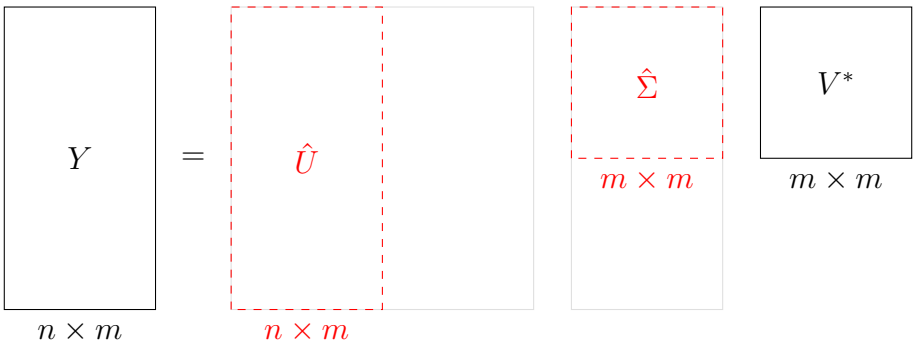


Figure 3.2: Economy SVD matrices shape.

These unnecessary components can be discarded without any loss of information, yielding the *economy* (or *thin*) SVD:

$$Y = \hat{U}\hat{\Sigma}V^* \tag{3.4}$$

where $\hat{U} \in \mathbb{R}^{n\times m}$ contains the columns of U associated with non-zero singular values[5], while $\hat{\Sigma} \in \mathbb{R}^{m\times m}$ contains the corresponding singular values. The matrix $V^* \in \mathbb{R}^{m\times m}$ remains unchanged. Therefore, the economy SVD provides the same information as the full SVD but with a reduced computational and memory cost,

[4] This is the typical situation in LES simulations: the number of spatial points can be of the order of 10^6 or higher, while the number of snapshots is usually of the order of 10^2–10^3.

[5] It is important to note that $\hat{U}$ is no longer a unitary matrix: its columns remain orthonormal ($\hat{U}^*\hat{U} = I$), but since $\hat{U}$ is no longer square, the relation $\hat{U}\hat{U}^* = I$ does not hold in general.

since only the m columns of U that can contribute to the decomposition of Y are computed and stored, instead of the full n columns.

Truncated SVD and the Optimality of POD

The economy SVD removes the columns of U that are associated with zero singular values, since they do not contribute to the reconstruction of the matrix Y. However, a further approximation can be obtained by exploiting the ordering of the singular values (3.3): since the singular values σ_j decrease monotonically, the contribution of each successive mode becomes smaller and may be discarded with only a limited loss of accuracy. By retaining only the first $r \leq m$ singular values and the corresponding left and right singular vectors, the *rank-r truncated SVD* is obtained:

$$Y \approx U_r \Sigma_r V_r^* \tag{3.5}$$

where $U_r \in \mathbb{R}^{n \times r}$, $\Sigma_r \in \mathbb{R}^{r \times r}$, and $V_r^* \in \mathbb{R}^{r \times m}$ are obtained by keeping only the first r columns of $\hat{U}$, the leading $r \times r$ block of $\hat{\Sigma}$, and the first r rows of V^*, respectively.

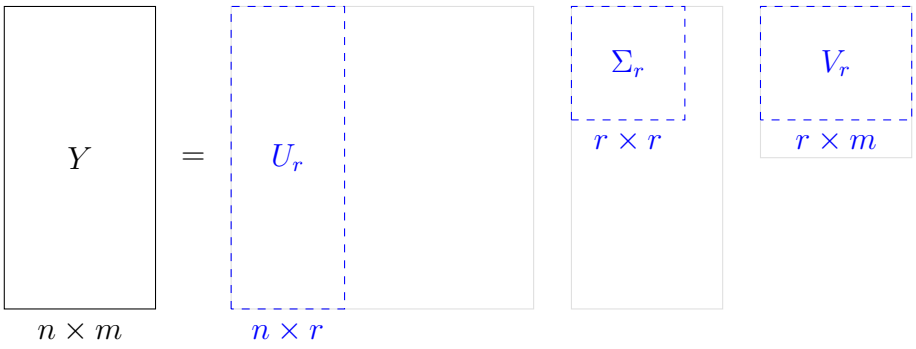


Figure 3.3: Truncated SVD matrices shape.

The quality of the truncated approximation can be measured using the *Frobenius norm*. For a matrix $A \in \mathbb{R}^{n \times m}$, the Frobenius norm is defined as

$$\|A\|_F = \sqrt{\sum_{j=1}^{n} \sum_{k=1}^{m} |A_{jk}|^2}, \tag{3.6}$$

and provides a measure of the "total energy" contained in the matrix. It can therefore be interpreted as the natural extension of the Euclidean norm from vectors to matrices.

For the matrix Y, the Frobenius norm can also be expressed in terms of its singular values as

$$\|Y\|_F^2 = \sum_{j=1}^{p} \sigma_j^2, \tag{3.7}$$

showing that the squared singular values represent the contribution of the corresponding modes to the total energy of the dataset.

The reconstruction error between the original matrix Y and its rank-r approximation Y_r can therefore be expressed as

$$\|Y - Y_r\|_F = \sqrt{\sum_{j=1}^{n} \sum_{k=1}^{m} |Y_{jk} - (Y_r)_{jk}|^2}, \tag{3.8}$$

which measures the difference between the original matrix Y and its truncated approximation Y_r.

The truncated SVD is not merely a convenient approximation; according to the Eckart-Young theorem [12], it is the *optimal* low-rank approximation of the matrix Y in the sense of the *Frobenius norm*. The truncation error is given exactly by

$$\|Y - Y_r\|_F = \sqrt{\sum_{j=r+1}^{p} \sigma_j^2} \tag{3.9}$$

This equation is particularly important in practice because it shows that the error introduced by truncating the decomposition at rank-r depends only on the discarded singular values. If the discarded singular values are small, the lost energy is negligible. This is what makes POD so powerful for flows with a few dominant structures (like a wind turbine wake): most of the energy sits in the first modes, and the rest can be safely discarded.

According to the Eckart–Young theorem, the truncated SVD provides the smallest possible reconstruction error among all rank-r approximations of Y. This optimality property is the basis of the optimality of POD: the first r POD modes capture the maximum possible amount of information, or energy, for a given number of retained modes.

Rank-1 expansion

An equivalent way to express the truncated SVD is as a sum of *rank-1* matrices. By using the diagonal structure of Σ_r, the product $U_r \Sigma_r V_r^*$ can be written as:

$$Y \approx U_r \Sigma_r V_r^* = \sum_{j=1}^{r} \sigma_j \, \boldsymbol{u}_j \boldsymbol{v}_j^* \tag{3.10}$$

where $\boldsymbol{u}_j \in \mathbb{R}^n$ and $\boldsymbol{v}_j \in \mathbb{R}^m$ are the j-th left and right singular vectors, respectively, and each outer product $\boldsymbol{u}_j \boldsymbol{v}_j^*$ is a rank-1 matrix. Each term in the sum represents the

contribution of the j-th singular mode to the reconstruction of Y, with its magnitude determined by the corresponding singular value σ_j. Larger singular values indicate that the corresponding modes contribute more significantly to the reconstruction of Y, whereas smaller singular values indicate a smaller contribution.

Making the dependence on space and time explicit, the decomposition (3.10) becomes:

$$\begin{aligned} Y(x,t) &= \sum_{j=1}^{m} \sigma_j \, \boldsymbol{u}_j(x) \, \boldsymbol{v}_j^*(t) \\ &\approx \sum_{j=1}^{r} \sigma_j \, \boldsymbol{u}_j(x) \, \boldsymbol{v}_j^*(t), \end{aligned} \tag{3.11}$$

where the dependence on x and t has been made explicit to highlight the spatial and temporal components of the decomposition. This is precisely the decomposition introduced in (3.1): the *POD modes* $\boldsymbol{u}_j(x)$ correspond to the left singular vectors of Y, while the corresponding *temporal coefficients* are given by $a_j(t_k) = \sigma_j v_j^*(k)$, where $v_j^*(k)$ denotes the k-th component of the j-th right singular vector associated with the snapshot at time t_k, and the singular values σ_j quantify the relative importance of the corresponding mode [13].

3.2.4 Computing POD: Eigendecomposition, Method of Snapshots and Efficient SVD Algorithms

The SVD formulation of POD is closely related to an eigenvalue decomposition of two correlation matrices constructed from the snapshot matrix Y. Starting from $Y = U\Sigma V^*$, it follows that the left singular vectors $\boldsymbol{u}_j$ and right singular vectors $\boldsymbol{v}_j$ satisfy the eigenvalue problems [6]

$$YY^*\boldsymbol{u}_j = \sigma_j^2 \boldsymbol{u}_j, \tag{3.12}$$

$$Y^*Y\boldsymbol{v}_j = \sigma_j^2 \boldsymbol{v}_j. \tag{3.13}$$

Therefore, the singular vectors of Y are the eigenvectors of the correlation matrices YY^* and Y^*Y, while the corresponding eigenvalues are the squared singular values σ_j^2.

The matrices $YY^* \in \mathbb{R}^{n\times n}$ and $Y^*Y \in \mathbb{R}^{m\times m}$ have different interpretations. The matrix Y^*Y describes the correlation in time between snapshots: each entry is given

[6] This follows directly from the SVD: substituting $Y = U\Sigma V^*$ into YY^* gives $YY^* = U\Sigma\Sigma^*U^*$, which is the eigendecomposition of YY^* with eigenvectors $\boldsymbol{u}_j$ and eigenvalues σ_j^2. An analogous argument holds for Y^*Y.

by

$$(Y^*Y)_{jk} = \boldsymbol{y}_j^* \boldsymbol{y}_k = \langle \boldsymbol{y}_j, \boldsymbol{y}_k \rangle_x, \tag{3.14}$$

where $\boldsymbol{y}_j$ represents the flow field at time t_j. Hence, $(Y^*Y)_{jk}$ measures the similarity between two snapshots over the whole spatial domain. Large values indicate snapshots with similar spatial structures, whereas small values indicate weak correlation. Similarly, YY^* represents the spatial correlation matrix, where each entry measures the correlation between the temporal signals associated with two spatial locations.

Since $Y \in \mathbb{R}^{n \times m}$ is typically a tall matrix in fluid mechanics applications ($n \gg m$ in LES computations), the two correlation matrices differ significantly in size. The matrix $YY^* \in \mathbb{R}^{n \times n}$ is generally too large to be formed and decomposed explicitly, whereas $Y^*Y \in \mathbb{R}^{m \times m}$ is much smaller. This observation leads to the *Method of Snapshots*, introduced by Sirovich [14], which computes the POD modes through the smaller temporal correlation matrix.

Instead of solving the eigenvalue problem associated with YY^*, the Method of Snapshots solves

$$Y^*Y\boldsymbol{v}_j = \sigma_j^2 \boldsymbol{v}_j, \tag{3.15}$$

providing the singular values and the right singular vectors. The spatial POD modes are then recovered from the relation

$$\sigma_j \boldsymbol{u}_j = Y\boldsymbol{v}_j, \tag{3.16}$$

or equivalently,

$$\boldsymbol{u}_j = \frac{1}{\sigma_j} Y\boldsymbol{v}_j. \tag{3.17}$$

Therefore, the Method of Snapshots does not define a different decomposition: it is an alternative computational procedure for obtaining the same POD modes when the number of snapshots is much smaller than the number of spatial degrees of freedom.

Although the Method of Snapshots is widely used and efficient for tall snapshot matrices, other approaches can be more suitable for large datasets. In particular, SLEPc provides several solvers for computing a singular value decomposition, retrieving only the dominant singular values and singular vectors without explicitly constructing the correlation matrices. This can significantly reduce the computa-

tional cost and memory requirements when only a limited number of dominant POD modes are required. In the present work, the thick-restart Lanczos method, denoted as `TRLANCZOS`, is used for this purpose. Its numerical implementation within the POD framework is described in Section 3.4.5.

3.2.5 Weighted Inner Products and Non-Uniform Grids

The POD formulation introduced above is based on the standard Euclidean inner product between snapshots. This choice is appropriate when the data points are uniformly distributed in the spatial domain (Ω_x), since the discrete inner product provides a direct approximation of the continuous one. In the continuous formulation, the spatial inner product between two snapshots is defined as

$$\langle \boldsymbol{y}_j, \boldsymbol{y}_k \rangle_x = \int_{\Omega_x} \boldsymbol{y}(x, t_j) \cdot \boldsymbol{y}(x, t_k) \, \mathrm{d}x, \tag{3.18}$$

However, in computational fluid dynamics simulations, and particularly in LES, the spatial grid is often non-uniform, with finer resolution in regions of high velocity gradients. In this case, the standard Euclidean inner product does not correctly represent the physical contribution of each region, as regions with a higher number of grid points are given a greater weight. This introduces a bias in the POD modes towards regions with finer spatial resolution. To account for the physical contribution of each region, a *weighted inner product* is therefore introduced.

$$\langle \boldsymbol{y}_j, \boldsymbol{y}_k \rangle_x = \boldsymbol{y}_j^T W_x \boldsymbol{y}_k, \tag{3.19}$$

where $W_x \in \mathbb{R}^{n \times n}$ is a symmetric positive definite weight matrix. In the simplest case, W_x is diagonal, with entries proportional to the local cell volumes associated with each spatial grid point, so that the discrete inner product consistently approximates the continuous integral (3.18) regardless of the local grid spacing.

In practice, the weighted POD can be computed by applying the standard SVD to the weighted snapshot matrix

$$\tilde{Y} = W_x^{1/2} Y \tag{3.20}$$

and then transforming the resulting modes back to the physical space. In this way, the orthogonality of the POD modes is defined according to the weighted inner product rather than the standard Euclidean one.

3.2.6 Limitations of Classical POD

POD provides an optimal low-rank representation of the data. For a given number of modes, the truncated representation minimises the reconstruction error with respect to the original dataset. This optimality is directly related to the ordering of the POD modes according to their energy contribution, so that the first modes capture the largest fraction of the total energy of the dataset. However, this does not imply that each mode represents a distinct dynamical process, frequency component, or characteristic timescale.

As a result, POD does not provide an explicit frequency decomposition. Different dynamical features may be represented by the same POD mode, making their individual contributions difficult to identify. Therefore, although POD efficiently captures the dominant energetic structures of a flow, it does not necessarily separate its different dynamical features. When information about specific frequencies is required, alternative formulations such as Spectral Proper Orthogonal Decomposition (SPOD) can be employed.

Another limitation of classical POD appears when analysing travelling or oscillating coherent structures. Since POD is based on spatial correlations, a structure moving through the domain is generally represented by several spatial modes with different phases rather than by a single travelling structure. In particular, a travelling structure is represented by a pair of stationary POD modes with similar spatial distributions and a phase shift between them in the advection direction. Consequently, the phase information associated with the motion of coherent structures is not directly preserved, which may complicate the interpretation of advected patterns in unsteady wakes.

These limitations motivate extensions of classical POD that include phase information in the modal decomposition. One of those approaches is the Hilbert Proper Orthogonal Decomposition (HPOD), which uses the Hilbert transform to construct a complex-valued representation of the flow field. The main principles of this approach are introduced below.

3.3 Hilbert POD

3.3.1 Motivation for HPOD

As discussed previously, classical POD may represent travelling coherent structures using a pair of spatial modes with different phases. Although this representation is optimal in terms of reconstruction error, the motion of the structure is distributed

between two modes rather than being directly represented by a single mode.

A more compact representation can be obtained by extending the decomposition to the complex domain. In this framework, a travelling wave can be represented by a single complex-valued mode whose amplitude and phase describe the temporal evolution of the coherent structure. This complex representation is obtained by applying the Hilbert transform to the original real-valued flow field, providing the basis for the Hilbert Proper Orthogonal Decomposition (HPOD).

As in the case of classical POD, HPOD modes are not spectrally pure and may contain contributions from different frequencies. Consequently, the decomposition remains optimal in terms of energy, but does not provide a complete separation of the different dynamics at different frequencies.

3.3.2 Hilbert Transform and Analytic Signal

To obtain a complex-valued representation of the flow field, the real-valued signal can be extended by means of its *analytic signal* by applying the Hilbert transform. For a one-dimensional temporal signal $s(t)$, the analytic signal is defined as:

$$\tilde{s}(t) = s(t) + i\,\mathcal{H}_t[s(t)], \tag{3.21}$$

where i is the imaginary unit and $\mathcal{H}_t$ denotes the Hilbert transform operator applied in time [11].

The Hilbert transform applied to the real-valued signal provides the imaginary component of the analytic signal. The Hilbert transform is defined as the Cauchy principal value[7] of the convolution of s(t) with the signal $1/\pi t$ according to:

$$\mathcal{H}[s(t)] = \frac{1}{\pi}\,\text{P.V.}\int_{-\infty}^{\infty} \frac{s(\tau)}{t-\tau}\,d\tau \tag{3.22}$$

A more intuitive interpretation can be obtained in the frequency domain, where the Hilbert transform corresponds to a phase shift of $\pi/2$ (either in advance or in delay, depending on the sign of the frequency component). This can be shown in the following equation using $\mathcal{F}$ to indicate the Fourier transform:

[7] The Cauchy principal value of a finite integral of a function f about a point c with $a \leq c \leq b$ is given by:

$$P.V.\int_a^b f(x)dx \equiv \lim_{\epsilon\to 0^+}\left[\int_a^{c-\epsilon} f(x)dx + \int_{c+\epsilon}^b f(x)dx\right]$$

and in this case is used to regularise the singular integral at $\tau = t$ by taking a symmetric limit around the singularity, ensuring the convergence of the Hilbert transform.

$$\mathcal{F}\{\mathcal{H}[s]\}(f) = -i\,\mathrm{sgn}(f)\,\mathcal{F}\{s\}(f) \tag{3.23}$$

which leads to the following expression for the analytic signal:

$$\tilde{s}(t) = \mathcal{F}^{-1}\left\{[1 + i\,\mathrm{sgn}(f)]\,\mathcal{F}\{s(t)\}\right\} \tag{3.24}$$

This expression shows that the analytic signal shares the same frequency content as the original signal for positive frequencies, whereas the content for negative frequencies is discarded.

However, in practice, the Hilbert transform is rarely evaluated directly from Eq.(3.22), since the singular integral is computationally expensive due to the singularity when $\tau = t$. Instead, it is commonly evaluated in the frequency domain using the so-called Fast Fourier Transform (FFT) followed by the Inverse Fast Fourier Transform (IFFT), as expressed in Eq. (3.24), to obtain the resulting analytic signal in the temporal domain.

The resulting analytic signal can be expressed in the polar form as follows:

$$\tilde{s}(t) = A(t)\,e^{i\phi(t)} \tag{3.25}$$

where $A(t) = |s(t)|$ and $\phi(t)$ represent, respectively, the instantaneous amplitude and instantaneous phase.

3.3.3 HPOD formulation

Once the dataset has been extended to the complex domain, the standard POD introduced in Section 3.2 can be applied to the resulting complex-valued data. The corresponding eigenvalue problem is written as:

$$\tilde{Y}\tilde{Y}^{*}\boldsymbol{\phi}_j = \sigma_j^2\boldsymbol{\phi}_j, \tag{3.26}$$

where $\tilde{Y}$ is the complex-valued snapshot matrix and $\boldsymbol{\phi}_j$ are the resulting complex-valued spatial modes. The singular value decomposition of $\tilde{Y}$ can then be written as:

$$\tilde{Y} = \sum_{j=1}^{m} \sigma_j \boldsymbol{\phi}_j \boldsymbol{v}_j^{*}, \tag{3.27}$$

where $\boldsymbol{v}_j$ denotes the j-th right singular vector. Making the temporal dependence explicit, the complex-valued velocity field can therefore be expressed as:

$$\tilde{\boldsymbol{u}}(\boldsymbol{x},t) = \sum_{j=1}^{m} \sigma_j \psi_j(t) \boldsymbol{\phi}_j(\boldsymbol{x}) \tag{3.28}$$

where $\psi_j(t_k) = v_j^*(k)$ is the temporal coefficient associated with the j-th spatial mode at the snapshot time t_k. Both ψ_j and $\boldsymbol{\phi}_j$ are complex-valued, allowing the temporal amplitude and phase of the travelling structures to be represented within the decomposition.

3.3.4 Treatment of edge effects

In practice, the analytic signal is obtained through the discrete Fourier transform. Because the Fourier transform implicitly assumes periodicity of the sampled signal, discontinuities may arise at the boundaries of the sampled signal, corrupting the phase estimation.

Several strategies have been proposed to reduce these edge effects. In the present work, a tapering operation based on a Hann window was applied to the first and last $T/8$ portions of the signal in order to enforce a smooth transition at the boundaries, forcing the signal to be zero at the edges (thereby imposing the periodicity of the signal) [15].

3.4 Numerical implementation of the POD framework

The Hilbert-POD analyses are performed using a dedicated Python framework developed to handle the large datasets generated by CFD simulations. The framework used in this work is based on a computational tool originally developed by Eman Bagheri [16] for the analysis of flow fields obtained from Nek5000 simulations. The original implementation was adapted and extended in the present work to process the data generated by OpenFOAM simulations.

This section describes the main aspects of the resulting computational implementation, with particular emphasis on how the POD methodology is applied to a large amount of data provided by the OpenFOAM fluid dynamics simulations. The complete procedure was implemented in Python, with selected code snippets provided in the following sections to illustrate the main computational steps.

3.4.1 General computational workflow

The implementation combines MPI communications for distributed memory parallelization, PETSc for the management of distributed matrices and vectors, and SLEPc for the parallel computation of the singular value decomposition (SVD). The overall workflow consists of the distribution of CFD data, the construction of the snapshot matrix, and the subsequent post-processing that leads to the final POD decomposition. The main steps of the implementation are described in the following sections.

3.4.2 MPI data distribution and domain decomposition

The POD framework is designed to handle large datasets by distributing the information and parallelising the workload. The parallel communication is based on the Message Passing Interface (MPI), which allows the computational domain and the associated flow data (such as the velocity field) to be distributed among multiple processes.

```
comm = MPI.COMM_WORLD
rank = comm.Get_rank()
size = comm.Get_size()
```

The same Python script is executed by all MPI processes: each rank executes the same program but it operates with its own portion of the data; `rank` identifies the individual process, while `size` denotes the total number of MPI processes involved in the computation.

However, some operations, such as the initial reading of the computational domain, are assigned to a single process and subsequently communicated to the other ranks. Therefore, `rank 0` reads the cell centre positions and volumes (extracted from the OpenFOAM dataset), and sends the information on the number of cells to all the other ranks by using a parallel operation called *broadcast*:

```
if rank == 0:
    V_loc_full  = np.asarray(readscalar(basename, "0", "V"), dtype=DT_REAL)
    C_full      = readvector(basename, "0", "C")
    nel_full       = int(V_loc_full.shape[0])

nel_full = comm.bcast(nel_full, root=0)
```

The use of a single process for the initial file reading avoids having all MPI processes to access the same large dataset simultaneously. A fully parallel reading strategy could be implemented by assigning different portions of the dataset to the MPI ranks. However, the current approach was retained because the file reading does not represent the main computational bottleneck of the analysis.

Considering N_c number of cells and N_p number of MPI processes, it is possible to distribute the number of cells between the available processes:

$$N_{\text{base}} = \left[\frac{N_c}{N_p}\right] \tag{3.29}$$

The eventual remaining number of cells

$$N_{\text{rem}} = N_c - N_{\text{base}} N_p \tag{3.30}$$

is distributed by assigning one additional cell to the first N_{rem} processes. The corresponding arrays of local size are then constructed:

```
base       = nel_full // size
extra      = nel_full %  size
cell_start = rank * base + min(rank, extra)
cell_end   = cell_start + base + (1 if rank < extra else 0)
n_local    = cell_end - cell_start

counts_cells = np.array([base + (1 if r < extra else 0) for r in range (size)], dtype
    =int)

displs_cells = np.zeros(size, dtype=int)
displs_cells[1:] = np.cumsum(counts_cells[:-1])
```

The array `counts_cells` specifies how many cells are assigned to each rank, while `displs_cells` specifies the position at which the corresponding portion starts in the global array.

Subsequent MPI collective operations such as `Scatterv` allow different amounts of data, such as the cell volume and the cell positions, to be sent to each process:

```
sendbuf_c       = np.ascontiguousarray(centroids_full, dtype=DT_REAL) if rank == 0
    else None
centroids_local = np.empty((n_local, 3), dtype=DT_REAL)
comm.Scatterv([sendbuf_c, counts_cells * 3, displs_cells * 3, MPI.FLOAT],
    centroids_local.ravel(), root=0)
```

An additional selection of the computational domain can be performed: the script allows the user to define an `inclusion_box` and, if required, one or more `exclusion_boxes`. Therefore, each rank applies these conditions to its local portion of the domain, generating a Boolean mask that identifies the cells that are retained for the POD analysis.

```
keep_elem_local = np.ones(nel_loc_ref, dtype=bool)

# Apply inclusion/exclusion criteria to local cell centres
...
```

This is particularly useful when the objective is to analyse coherent structures in a specific region of the flow, without including the entire domain.

The same domain decomposition is subsequently used to distribute the velocity field associated with each snapshot among the MPI ranks. For each snapshot, the local values of three velocity components can be arranged into a local vector, and each local vector is then used to populate the distributed snapshot matrix. The velocity field is read by `rank 0` and distributed between ranks according to `counts_cells` and `displs_cells`, sending to each rank only the velocity values associated with its assigned cells.

```
if rank == 0:
    field_full = readvector(basename, time_str, "U")

recvbuf = np.empty(n_local * 3, dtype=DT_REAL)
comm.Scatterv([field_full, counts_cells * 3, displs_cells * 3, MPI.FLOAT], recvbuf,
    root=0)
field = recvbuf.reshape(n_local, 3).T
```

The received data are initially stored in a one-dimensional array that contains the three velocity components for each cell. The array is then reshaped into a matrix with dimensions $(N_{local}, 3)$, where each row corresponds to a cell and the three columns contain the corresponding x, y, and z velocity components. The transpose operation changes its organisation to $(3, N_{local})$:

$$\begin{bmatrix} u_{x,1}^{(k)} & u_{x,2}^{(k)} & \cdots & u_{x,N_{local}}^{(k)} \\ u_{y,1}^{(k)} & u_{y,2}^{(k)} & \cdots & u_{y,N_{local}}^{(k)} \\ u_{z,1}^{(k)} & u_{z,2}^{(k)} & \cdots & u_{z,N_{local}}^{(k)} \end{bmatrix} \tag{3.31}$$

where $u_{x,n}^{(k)}$, $u_{y,n}^{(k)}$ and $u_{z,n}^{(k)}$ correspond to the three velocity components of the n-th local cell, and k denotes the k-th snapshot.

3.4.3 Snapshots matrix assembly

The organisation shown at the end of Section 3.4.2 is particularly convenient because the three rows can be concatenated to form a single vector containing all the spatial degrees of freedom at a given time instant. Repeating this operation for all snapshots allows the snapshot matrix to be constructed with the spatial degrees of freedom along its rows and the different time instants along its columns. Therefore, the resulting vector stacks the three velocity components at time step k:

$$\mathbf{a}^{(k)} = \begin{bmatrix} \mathbf{u}_x^{(k)} \\ \mathbf{u}_y^{(k)} \\ \mathbf{u}_z^{(k)} \end{bmatrix}$$

The resulting matrix is defined as:

$$\mathbf{A} = \begin{bmatrix} | & | & & | \\ \mathbf{a}^{(1)} & \mathbf{a}^{(2)} & \dots & \mathbf{a}^{(N_t)} \\ | & | & & | \end{bmatrix}$$

Therefore, each column represents one complete velocity snapshot, whereas each row contains the temporal evolution of a specific spatial degree of freedom. In the script, this is implemented as follows:

```
for name in variables_to_read:
    comp_idx = variables_to_read[name]
    comp = field[comp_idx, keep_elem_local].astype(DT_REAL, copy=False)
    flats.append(comp)

snap_local = np.concatenate(flats); del flats
```

The local snapshot vector is then weighted by the square root of the cell volumes:

```
snap_w = snap_local * bm1sqrt_vec
A.setValues(rows_owned, np.array([k], dtype=IDX_DTYPE), snap_w.reshape(-1,1), addv=
    PETSc.InsertMode.INSERT_VALUES)
```

where

```
sqrtB_flat_sub = np.sqrt(massScaleFactor * B_sub).astype(DT_REAL, copy=False)
```

In the parallel implementation, each process stores only the rows associated with its local degrees of freedom; the `setValues` operation therefore inserts only the portion corresponding to the spatial degrees of freedom owned by the current MPI process, avoiding the need to store the full matrix on a single process.

The snapshot matrix is weighted according to the cell volumes to account for the non-uniformity of the computational mesh, as described in Section 3.2.5. The weighting operation can be expressed mathematically as:

$$\mathbf{A}_w = \mathbf{M}^{1/2}\mathbf{A}$$

where $\mathbf{M}$ is the diagonal mass matrix, where each diagonal element is associated with the cell volumes. In the implementation, the matrix $\mathbf{M}^{1/2}$ is not explicitly assembled but instead a weighting vector is built as follows:

```
bm1sqrt_vec = np.concatenate([sqrtB_flat_sub] * len(variables_to_read)).astype(
    DT_REAL, copy=False)
```

3.4.4 Temporal processing and Hilbert transform

Before performing the POD decomposition, the framework allows to filter the temporal signal using a second-order Butterworth band-pass filter[8], in order to isolate the frequency range of interest and remove unwanted low or high-frequency content. A forward and backward filtering procedure is used to obtain a zero-phase filtered signal, thereby avoiding phase distortion introduced by the filtering operation.

```
if filtering:

    dt_eff = float(np.median(time_array[1:] - time_array[:-1])) if T0 > 1 else 1.0
    fs  = 1.0 / dt_eff
    nyq = 0.5 * fs

    bhp, ahp = _biquad_highpass(fmin_hz, fs, Q=filter_Q, dtype=dtype) if have_hp else
     (None, None)
    blp, alp = _biquad_lowpass(fmax_hz, fs, Q=filter_Q, dtype=dtype)  if have_lp else
     (None, None)

    _process_half(slice(0, T0), arr_local, S_block, have_hp, have_lp, bhp, ahp, blp,
    alp, apply_taper=apply_edge_taper)
```

After filtering, the temporal mean can be removed independently for each spatial degree of freedom. This operation is performed row-wise on the snapshot matrix and results in a fluctuating field centred around zero:

```
if remove_mean_after_filter and S_local > 0:
        arr_local -= arr_local.mean(axis=1, keepdims=True)
```

For the Hilbert-POD analysis, the real-valued temporal signals are subsequently transformed into analytic signals using the Hilbert transform. For a generic velocity signal $u(t)$, the corresponding analytic signal is defined as

$$\tilde{u}(t) = u(t) + i\mathcal{H}[u(t)]$$

The resulting complex-valued signals are then used as input for the POD decomposition described in Section 3.3.3.

In the implementation, the Hilbert transform is applied independently to each row of the snapshot matrix and is performed in the frequency domain using the Fast Fourier Transform (FFT). Before applying the FFT, a temporal taper can be applied to the signal near the boundaries of the temporal window to impose the periodicity of the signal and limit the spectral leakage.

[8]The Butterworth filter is a frequency-selective filter characterized by a maximally flat magnitude response in the passband. The forward-backward filtering procedure used in this implementation provides a zero-phase response, avoiding phase shift that could alter the temporal evolution of the flow structures.

```
H = np.zeros(N, dtype=np.complex64)
H[0] = 1.0
if N % 2 == 0:
    H[1:N//2] = 2.0
    H[N//2] = 1.0
else:
    H[1:(N+1)//2] = 2.0
```

As stated above, the Hilbert transform is implemented in the frequency domain using the FFT: each temporal signal is transformed into the frequency domain, where H suppresses the negative-frequency components and doubles the positive-frequency components. The transformation is then completed by applying the IFFT, which provides the corresponding analytic signal in the time domain:

```
fft_plan()
X_buf *= H
ifft_plan()
```

The resulting complex-valued matrix is then used for the subsequent Hilbert-POD decomposition.

3.4.5 POD decomposition and extraction of the modes

Once the snapshot matrix $\mathbf{A}_w$ has been assembled and processed, the POD decomposition is performed according to the SVD formulation introduced in Section 3.2.4.

$$\mathbf{A}_w = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$$

In the implementation, two alternative approaches are available: the Method of Snapshots (MoS) and the direct, iterative computation of the SVD using the SLEPc-based TRLANCZOS solver (the latter is the approach used in the present work).

For the Method of Snapshots, the temporal correlation matrix is assembled from the distributed portions of the snapshot matrix according to

$$\mathbf{C} = \mathbf{A}_w^T\mathbf{A}_w$$

Since the rows of $\mathbf{A}_w$ are distributed among MPI ranks, each process computes the contribution associated with its local rows:

```
blk = arr_eff[rs:re, :].astype(acc_dtype, copy=False)
C_local += blk.conj().T @ blk
```

The local contributions are subsequently combined to obtain the global correlation matrix. The eigenvalues and eigenvectors are then computed on a single rank

and broadcast to all processes. The spatial modes are finally reconstructed according to:

$$\boldsymbol{\phi}_j = \frac{\mathbf{A}_w \mathbf{v}_j}{\sigma_j}$$

evaluated for each column, avoiding the formation of the full matrix product explicitly.

Alternatively, the SLEPc SVD solver can be used to compute the decomposition of the matrix $\mathbf{A}_w$ directly through an iterative method (thick-restart Lanczos, TRLANCZOS):

```
svd = SLEPc.SVD().create(comm=comm)
svd.setType(SLEPc.SVD.Type.TRLANCZOS)
svd.setDimensions(nsv=nsv_request)
svd.setOperator(A)
svd.solve()
```

In this approach, the correlation matrix does not need to be explicitly assembled; instead, SLEPc directly computes the required singular values and the corresponding left and right singular vectors through `svd.getSingularTriplet()`.

Once computed, each spatial mode is written back to disk in physical units by undoing the mass-matrix weighting and separating the modes into their three velocity components. The resulting fields are then saved in a format compatible with the OpenFOAM structure, allowing the POD modes to be visualized and further analysed using ParaView.

3.5 Validation of HPOD on a Cylinder Wake

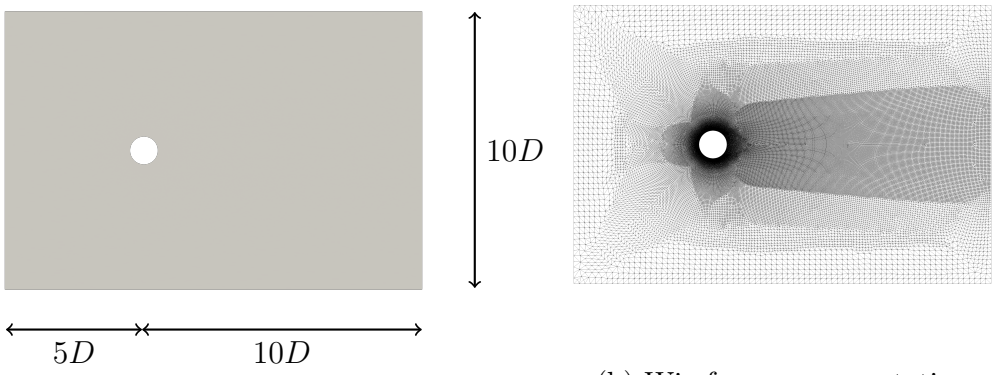
Before applying HPOD to the more complex floating offshore wind turbine wake, the method is validated on the canonical flow past a circular cylinder. The cylinder wake is characterized by a well-known periodic vortex-shedding behaviour, making it a suitable case for assessing the ability of the developed Python script implementation to identify coherent oscillatory structures. In this work, the existing HPOD Python script was adapted to process OpenFOAM output data and its results were compared with reference results from the literature.

3.5.1 Cylinder Flow Configuration

The fluid dynamics problem considered is a well-known and well-studied benchmark widely discussed in the literature: the vortex shedding caused by the flow past a

circular cylinder with the main axis perpendicular to the undisturbed stream velocity U_0.

The cylinder has a diameter of $D = 1\,\mathrm{m}$. The computational domain extends from $-5D$ to $10D$ in the streamwise direction, from $-5D$ to $5D$ in the crosswise direction, and from 0 to $3.14D$ in the spanwise direction. The computational mesh consists of approximately 4.0×10^6 cells, most of which are hexahedral.



(a) Surface representation

(b) Wireframe representation

Figure 3.4: Computational domain of the cylinder test case.

The flow is characterised by a uniform inlet velocity $U_0 = 1\mathrm{m/s}$ and a Reynolds number of $Re = 100$, corresponding to a kinematic viscosity of $\nu = 1 \times 10^{-2}\mathrm{m}^2/\mathrm{s}$.

At the inlet, a uniform velocity is prescribed, with a zero gradient condition for pressure. At the outlet, the pressure is uniform and fixed to $p = 0$, while a zero-gradient condition is applied to the velocity. A no-slip condition is imposed on the cylinder surface, with a zero-gradient pressure condition. Slip conditions are applied at the upper and lower boundaries, while periodic conditions are imposed in the spanwise direction.

From the literature, the vortex shedding of a flow around a cylinder at small Reynolds numbers is characterised by the Strouhal number:

$$St = \frac{f_s D}{U_0} \tag{3.32}$$

where f_s is the vortex shedding frequency, D is the cylinder diameter, and U_0 is the free-stream velocity. For a circular cylinder at $Re = 100$, $St \approx 0.2$ is expected. For the present case, this corresponds to a vortex shedding period of $T = 5\mathrm{s}$. This characteristic time scale is important for selecting the temporal resolution of the simulation and the subsequent data sampling. In particular, the sampling interval should be sufficiently smaller than the shedding periods to resolve the dominant

flow dynamics. In the present case, a sampling interval of $\Delta t_{sample} = 0.2$s was used, corresponding to 25 samples per shedding period. The timestep is also constrained by the Courant-Friedrichs-Lewy (CFL) number, which measures the ratio between the distance travelled by the flow during one timestep and the local cell size. It is defined as:

$$CFL = \frac{U\Delta t}{\Delta x} \tag{3.33}$$

Therefore, the simulation timestep used was set to $dt = 0.002$s to maintain a Courant number below one.

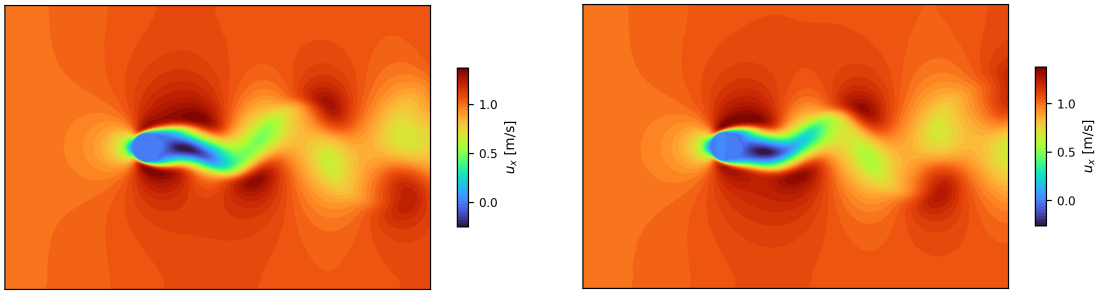


Figure 3.5: Vortex shedding behaviour of cylinder wake.

Before applying the HPOD, the wake must be sufficiently developed. To achieve this, the simulation was run for sufficiently long time. In the present work, a minimum simulation time of 10 shedding periods was considered sufficient for the wake to reach a developed state.

Figure 3.5 shows the instantaneous snapshot of the streamwise velocity field after the wake has developed. The alternating pattern of vortices downstream of the cylinder is clearly visible, confirming the expected vortex-shedding behaviour.

3.5.2 HPOD Implementation and Post-Processing

Once the fluid dynamics simulation is completed, the HPOD is applied using the Python script developed for the present work. The OpenFOAM simulation produces a series of time folders containing the flow fields at each time step (snapshot). The numerical framework is described in detail in Section 3.4, but its main steps are recalled here. At the first time step, rank 0 reads the cell positions of the entire domain and distributes this information among the available ranks. The snapshot matrix is then assembled, the temporal mean removed, and the Hilbert transform is applied to obtain its complex-valued representation. Finally the POD is performed on the

complexified data. The resulting HPOD modes are written as new OpenFOAM-like cases, allowing the spatial modes to be visualised directly in ParaView.

3.5.3 HPOD Results

Using the settings described previously, the first three HPOD modes are analysed to investigate the flow fluctuations. The main purpose of this test case is to demonstrate the ability of HPOD to identify advected coherent structures. The wake past a cylinder provides a simple example for this purpose. Figure 3.6 shows the first three spatial modes obtained by applying HPOD in time. The real and imaginary parts of each complex-valued mode are shown in the left and right columns, respectively.

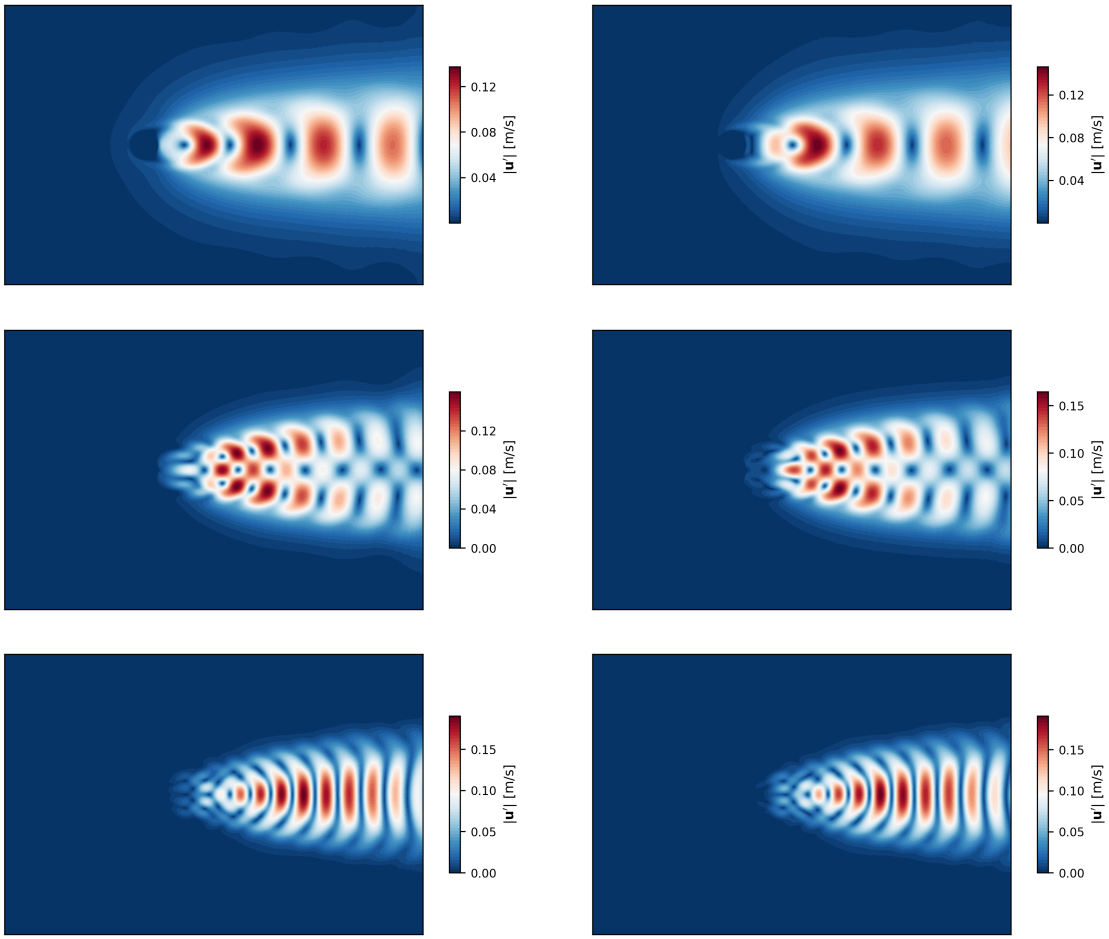


Figure 3.6: First three spatial HPOD modes of the cylinder wake. The real parts are shown in the left column and the imaginary parts in the right column.

The first three modes provide information about the spatial distribution of the dominant velocity fluctuations in the wake. The colour represents the value of the corresponding mode at each location, which can be positive or close to zero. These

figures should not be interpreted as instantaneous flow fields at a particular time. Instead, each mode represents a spatial structure obtained from the decomposition of the complete dataset and describes one of the dominant contributions to the fluctuating velocity field, while its corresponding temporal coefficient determines how this pattern evolves in time.

An important feature of HPOD is the presence of both real and imaginary parts. As explained in Section 3.3.2, since the analytic signal is constructed by applying the Hilbert transform in time, these two parts form a quadrature representation of the oscillatory dynamics, with a phase difference of approximately $\pi/2$. For a travelling structure, this phase difference can also appear as a spatial displacement of the structure, providing information about its propagation.

The HPOD modes are ordered according to their energy content, with the first modes containing the largest fraction of the fluctuation energy represented by the analysed modes. The energy associated with each mode is related to its corresponding eigenvalue, and the sum of the eigenvalues represents the total captured energy of the fluctuating field. For the present case, the first three modes contain 94.91 %, 3.38 %, and 1.56 % of this reference energy, respectively. Therefore, the first three modes contain approximately 99.84% of the energy represented by the ten modes considered for this analysis.

These results show that most of the energy represented by the analysed HPOD is concentrated within the first few modes: the first three modes account for more than 99% of the reference energy, indicating that they capture the dominant energetic contributions among the analysed modes of the cylinder wake.

HPOD mode	Energy content (%)	Cumulative energy content (%)
1	94.91	94.91
2	3.38	98.28
3	1.56	99.84
4	0.12	99.96
5	0.03	99.99

Table 3.1: Energy content of the first five HPOD modes among the ten considered for the analysis.

Focusing on the first mode, the real part identifies the main spatial regions associated with the dominant fluctuations in the cylinder wake. However, the imaginary part is also needed to describe the phase evolution of the oscillatory structure. To better understand the physical meaning of this mode, its streamwise and crosswise velocity components are shown in Figure 3.7.

Figure 3.7 shows the streamwise and crosswise components of the first HPOD mode. The streamwise component describes the fluctuations in the direction of the incoming flow, while the crosswise component describes the fluctuations perpendicular to it. For the streamwise component, positive values correspond to fluctuations in the downstream direction, while negative values correspond to fluctuations in the upstream direction. The alternating regions of positive and negative fluctuations are associated with the periodic vortex-shedding process. The crosswise component provides complementary information about the lateral motion induced by the vortices.

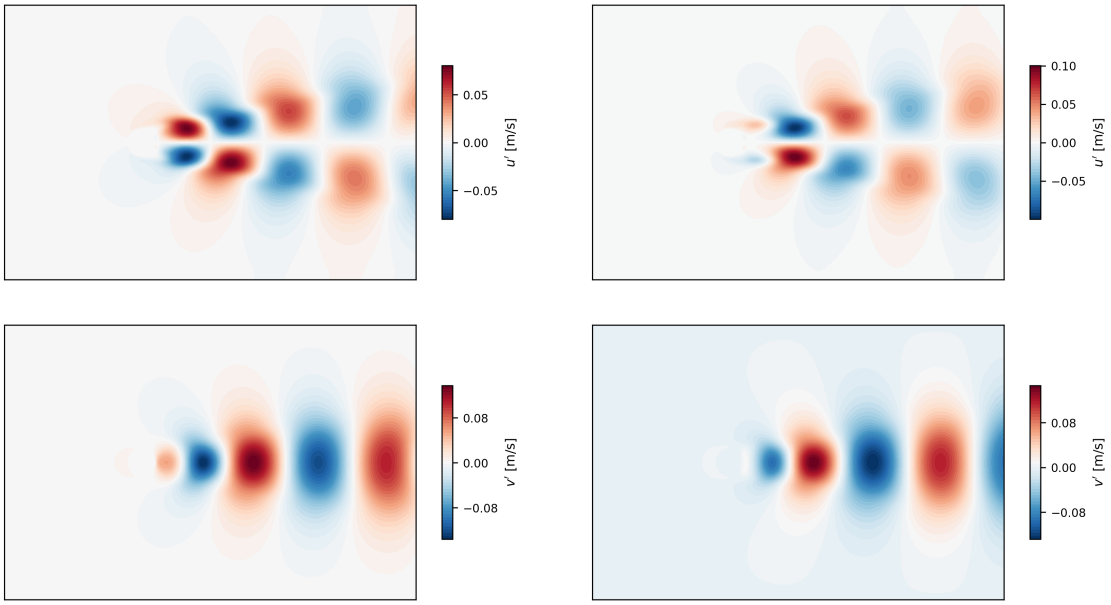


Figure 3.7: Components of the first HPOD mode. The top panels show the real (left) and imaginary (right) parts of the HPOD mode in the streamwise direction, while the bottom panel shows the corresponding components in the crosswise direction.

An important observation is that the first complex HPOD mode contains the information that would be distributed between the first two real-valued POD modes. Indeed, in classical POD, the periodic vortex-shedding process is typically represented by two spatial modes with a phase difference. One of the main advantages of using HPOD is that it combines these two quadrature components into a single complex mode, providing a more compact representation of the travelling structure.

Chapter 4

Wind Turbine Simulations

4.1 Simulation setup

To investigate the effect of platform-induced surge motion on the wake dynamics of the NREL5MW wind turbine, a set of fluid dynamics simulations was performed using the Large-Eddy Simulation (LES) approach and the Actuator Line Method. The simulations were carried out using the SOWFA-6 framework within OpenFOAM, specifically designed to perform fluid dynamics simulations of wind turbines. The numerical setup was kept unchanged among the different cases, while the rotor motion was varied to investigate the influence on the development and recovery of the turbine wake under surge motion.

The main characteristics of the numerical setup are described in the following sections, which cover the computational domain, the actuator-line configuration, the inlet flow, and turbulent conditions.

The present study considers a single floating offshore wind turbine and no additional interactions with other turbines, therefore the analysis focuses on the wake generated by an isolated turbine. Moreover, only the rotor blades are represented through the ALM, while the hub, nacelle, and tower are not considered.

4.1.1 Computational domain and mesh

Fig. 4.1 shows a schematic representation of the computational domain, which extends $24D$ in the streamwise direction (x) and $8D$ in both the spanwise and vertical directions, where $D = 126\,\mathrm{m}$ is the rotor diameter. The rotor centre is located at the centre of the domain in the vertical and spanwise directions. The inlet boundary is located $6.5D$ upstream of the rotor, while the outlet boundary extends $17.5D$ downstream.

At the inlet boundary, a uniform inflow with an unperturbed velocity of $U_0 = 9.6\,\mathrm{m/s}$ is prescribed. The outlet boundary is treated with a zero-gradient condition for the velocity ($\partial \mathbf{U}/\partial n = 0$), allowing the wake to leave the computational domain without imposing its velocity at the outlet. For the pressure, a fixed value of $p = 0$ is imposed at the outlet, while a zero-gradient condition is applied to the pressure at all other boundaries . The lateral boundaries are treated as free-slip boundaries ($U \cdot \mathbf{n} = 0$), thereby preventing the development of non-physical artificial boundary layers along the lateral domain boundaries.

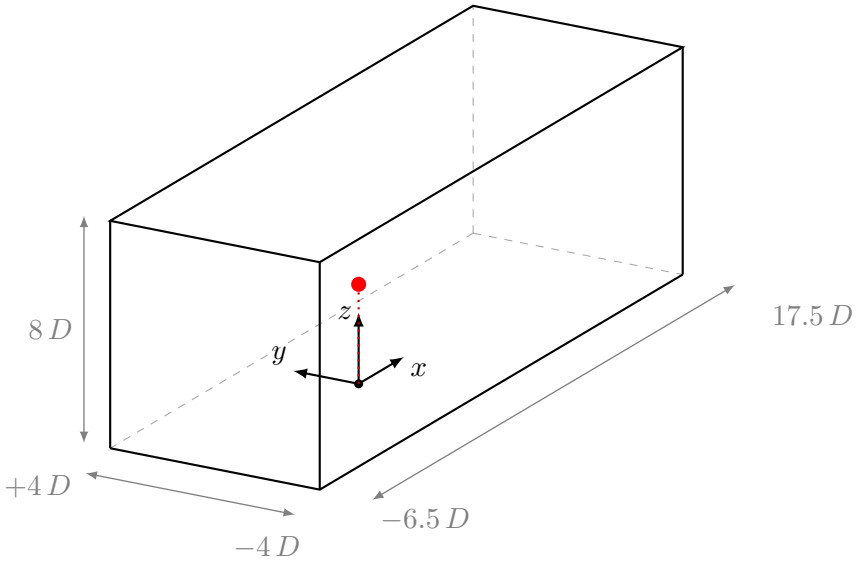


Figure 4.1: Computational domain of the wind turbine.

The computational domain is discretised using an initial isotropic cell size of 6 m. Three successive levels of refinement are introduced to adequately resolve the flow structure in the downstream region of the turbine wake and around the blades. The outer refinement region extends from $-3D$ to $17.5D$ in the streamwise direction and has a radius of $1.5D$, while the intermediate region extends from $-1.5D$ to $13.5D$ with a radius of $1D$. The innermost region, which extends from $-0.75D$ to $3D$ with a radius of $0.75D$, provides the highest resolution around the blades. These refinements reduce the cell size to 1.5 m in the wake region and to 0.75 m around the blades. The resulting mesh consists of more than 75 million cells. The refinement strategy is designed to provide a higher resolution where strong velocity gradients are expected, allowing the main features of the flow to be adequately captured. Moreover, the use of smaller cells is also advantageous from the ALM perspective, as it allows the use of a smaller ϵ to spread the point forces over a more confined region, providing a more accurate representation of the blade forces.

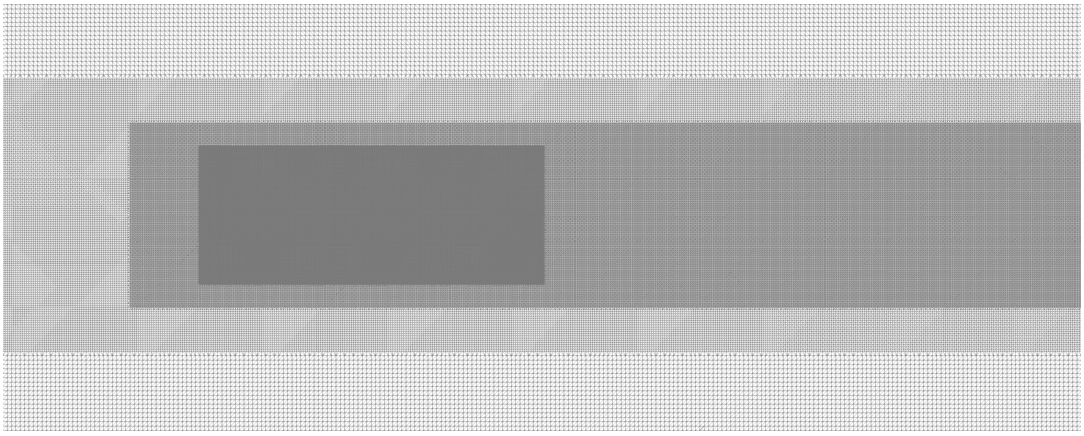


Figure 4.2: Discretized domain with three levels of refinement.

4.1.2 Turbine and actuator-line configuration

The simulations employ the NREL 5 MW reference wind turbine as the baseline rotor configuration, with a rotor diameter of $D = 126$m.

The blades are represented using the Actuator Line Method described in Section 2.2. Each blade is discretised into 32 actuator points distributed along the blade span, providing the spatial resolution required to describe the aerodynamic loading.

According to the guidelines reported in [17], the ratio between the actuator-point spacing and the local grid resolution should satisfy the following:

$$\Delta_b/\Delta_{grid} \geq 1.5 \tag{4.1}$$

where Δ_b is the distance between two consecutive actuator points and Δ_{grid} is the cell dimension in the blade region. For the present configuration, considering the cell dimension of 0.75 m and 32 uniformly distributed actuator points along the 63 m blade span, the resulting actuator point spacing is approximately 2.03 m, corresponding to $\Delta_b/\Delta_{\text{grid}} \approx 2.71$, which is consistent with the recommended criterion.

Moreover, a variable Gaussian force distribution is employed to project the aerodynamic forces from the actuator points onto the computational grid. In the present work, its width is varied along the blade span according to the local chord length, allowing the spatial distribution of the aerodynamic forces to account for the variation of the blade geometry along the span.

According to Jha et al. [17], the Gaussian width should be sufficiently large with respect to the local grid resolution to avoid numerical instabilities. It is recommended to ensure that the following relation is satisfied:

$$\epsilon \geq 2\Delta_{\text{grid}}. \tag{4.2}$$

In the present simulations, a slightly more conservative lower limit is adopted, with $\epsilon_{min} = 2.1\Delta_{grid}$. This ensures that the Gaussian force distribution is adequately represented by the computational grid, reducing the risk of instabilities. An upper limit of $\epsilon_{\max}$ is also imposed to prevent excessive spreading of the body force over the computational domain.

The Gaussian width is then determined locally along the blade by scaling the local chord length with a multiplicative coefficient $\epsilon_s = 0.5$ and applying the previously defined lower and upper limits. The resulting distribution is given by:

$$\epsilon = \max\left[\min\left(\epsilon_s c, \epsilon_{\max}\right), \epsilon_{\min}\right]. \tag{4.3}$$

where c is the local chord length of the blade. The resulting variation of ϵ along the blade span is shown in Fig. 4.3

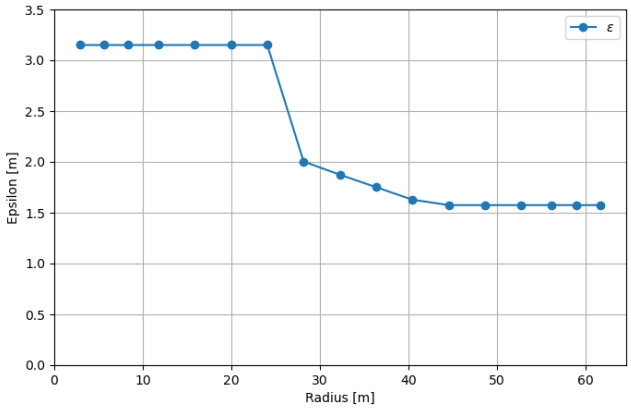


Figure 4.3: Variable blade ϵ.

An integral interpolation method is employed to obtain the local velocity experienced by the blade elements, while the resulting aerodynamic forces are projected onto the computational grid along the direction of local relative velocity.

4.1.3 Flow and turbulent conditions

A uniform inflow velocity of $U_0 = 9.6\,\mathrm{m/s}$ is prescribed at the inlet. The kinematic viscosity is set to $\nu = 1.0 \times 10^{-5}\,\mathrm{m^2/s}$, resulting in a Reynolds number of $Re_D = U_0\,D/\nu = 1.21 \times 10^8$.

The uniform inflow profile does not account for the mean velocity variation associated with an Atmospheric Boundary Layer (ABL), which is characterized by a logarithmic mean velocity profile and large coherent structures. However, the incoming turbulence level is prescribed through a turbulent kinetic energy at the inlet

of $k_{inlet} = 1.38\,\mathrm{m}^2/\mathrm{s}^2$. Assuming isotropic turbulence, it is possible to retrieve the turbulent intensity of the flow:

$$TI = \frac{\sqrt{2k/3}}{U_0} = 10\% \tag{4.4}$$

The simulations are performed using the Large-Eddy Simulation (LES) approach with the Smagorinsky model for the subgrid-scale turbulence. The large, energy-containing turbulent structures are explicitly resolved, whereas the effects of the smaller unresolved scales are modelled. The model coefficients adopted in the present simulations are $C_k = 0.094$ and $C_e = 1.048$, consistent with the values adopted in previous LES studies of wind-turbine wakes [18]. The filter scale used in the present work is defined as

$$\Delta = V_{cell}^{1/3} \tag{4.5}$$

where V_{cell} is the cell volume. Since an isotropic mesh is employed, the filter length scale coincides with the cell dimension.

4.1.4 Numerical setup

The simulations are performed using the in-house `pimple_ALMAdvanced` solver of SOWFA, which is based on the PIMPLE algorithm and includes the Actuator Line Method implementation used in the present work.

The simulation time step is set to $\Delta t = 0.012\,\mathrm{s}$. The time step is selected considering the maximum displacement of the actuator points during one time step; in particular, the blade tip represents the most restrictive condition since it experiences the highest tangential velocity. For the present turbine, with a rotational speed of $9.1552\,\mathrm{rpm}$ and a rotor radius of $R = 63\,\mathrm{m}$, the corresponding tip speed is

$$U_{\mathrm{tip}} = \Omega R \approx 60.4\,\mathrm{m/s}. \tag{4.6}$$

To limit the displacement of the actuator points to less than one grid cell during a single time step, as reported in [17], the following condition is considered:

$$U_{tip}\Delta t < \Delta_{grid} \tag{4.7}$$

With $\Delta_{grid} = 0.75\,\mathrm{m}$, this results in a $\Delta t < 0.0124\,\mathrm{s}$ limit, making the selected value of $\Delta t = 0.012\,\mathrm{s}$ consistent with the adopted spatial and temporal discretisation.

The selected time step also results in a Courant number[9] below unity throughout the simulation. In particular, a maximum value of approximately $CFL = 0.32$ is

obtained, which also fulfils the CFL constraint.

Before performing the simulations used for the POD and wake analyses, a preliminary simulation is carried out using a larger time step but maintaining the Courant number below the imposed limit of $CFL = 0.75$. This simulation is used to allow the flow to develop and the turbine power to reach a statistically stationary state. After approximately $t = 250\,\mathrm{s}$, the turbine power reaches a stable value and the resulting flow field is then used as the initial condition for the subsequent simulation cases.

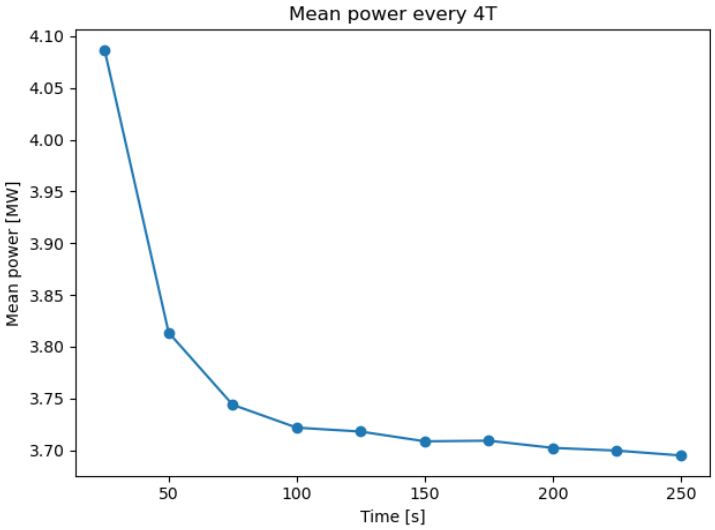


Figure 4.4: Evolution of the rotor power during the preliminary simulation.

4.1.5 Simulation cases

The dynamic motions of a Floating Offshore Wind Turbine (FOWT) are strongly influenced by the ocean motion, the type of the platform foundation and the mooring system with its anchoring configuration. The dynamic response of a FOWT can involve several degrees of freedom, among which surge and pitch motions can significantly affect the wake dynamics. In the present work, the surge motion is isolated and its effects on the wake dynamics are investigated.

[9]The Courant-Friedrichs-Lewy (CFL) number is a dimensionless value used to measure how the fluid moves across a grid cell in a single time step and it is used to determine if the simulation time step is safe and accurate. It is defined as follows:

$$CFL = \frac{U\,\Delta t}{\Delta x} \tag{4.8}$$

where U represents the local flow velocity magnitude, Δt the time step size and Δx the spatial grid cell size. It is a good practice to keep the CFL number below unity, because that would mean that the fluid moves less than one cell per time step. This is required for stability in explicit numerical solvers.

Following the frequency range suggested in Ref. [19] and based on the results reported in Ref. [20], two particular surging frequencies are investigated: a relatively low frequency, $St = 0.39$, and a higher frequency of $St = 0.91$, where the Strouhal number[10] is defined as:

$$St_s = \frac{f_s D}{U_0} \tag{4.9}$$

For a real-scale wind turbine, the maximum amplitude of its surge motion has been found to reach approximately $5\% - 12.5\%$ of the rotor diameter [21, 20]. Accordingly, a relatively large dimensionless surge amplitude, defined as $A^* = A_s/D = 0.08$, is prescribed, to facilitate a systematic investigation of frequency-dependent wake behaviour while remaining within the bounds of physically realistic motion. The amplitude is kept fixed across all frequencies to isolate the effect of the surge frequency on the wake features.

Based on these considerations, three simulation cases are considered: one static case used as a reference and two harmonic surge cases with identical amplitude A_s and different surge frequencies. The prescribed harmonic surge motion in the streamwise direction imposed to the wind turbine (with an initial offset of zero) is defined as:

$$x(t) = A_s \sin(2\pi f_s t) = A_s \sin(\omega_s t) \tag{4.10}$$

where f_s and A_s represent, respectively, the surge motion frequency and amplitude, and $\omega_s = 2\pi f_s$ is the angular surge frequency.

Case	Rotor motion	A_s [m]	f_s [Hz]	ω_s [rad/s]	St_s
Static	None	–	–	–	–
Surge 1	Harmonic	10.08	0.0297	0.1867	0.39
Surge 2	Harmonic	10.08	0.0693	0.4356	0.91

Table 4.1: Simulation cases considered in the present work.

4.2 Wake characteristics of the static rotor

The fixed-bottom configuration is first analysed to characterise the baseline wake behaviour in the absence of platform motion. This case is used as a reference for the

[10]The Strouhal number is a dimensionless frequency that is used to describe oscillating flow mechanisms.

subsequent analysis of the moving platform configurations and allows the intrinsic turbine wake behaviour to be distinguished from the effects associated with the rotor surge motion.

The wake is analysed within the three-dimensional sampling region introduced for the POD analysis. Particular attention is paid to the streamwise evolution of the wake velocity deficit and to the identification of the main coherent structures that develop downstream of the rotor. The spatial evolution of the wake is considered at different downstream locations to characterise its development and recovery.

4.2.1 Wake structure and velocity field

The streamwise velocity field of the static rotor is first analysed. Figure 4.5 shows a horizontal xy plane section that passes through the rotor hub, with the computational domain extending from $-1.5\,D$ upstream of the rotor to $17.5\,D$ downstream. The colour scale indicates the streamwise velocity component u_x. These regions correspond to the velocity deficit generated by the turbine and the reduction in the flow momentum, providing a first indication of how the wake develops downstream.

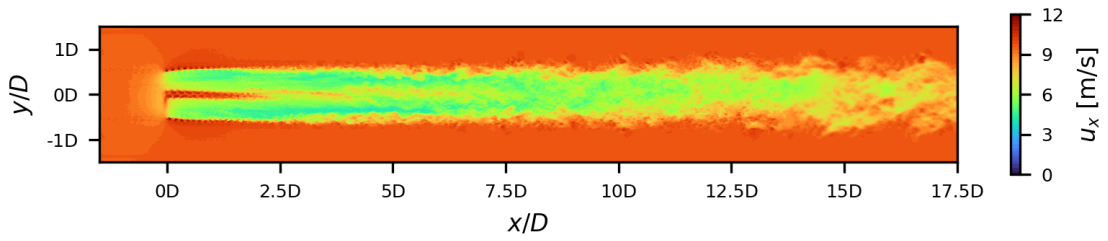


Figure 4.5: Velocity field at $t = 600$ s.

A pronounced velocity deficit is observed immediately downstream of the rotor, corresponding to the momentum extracted by the wind turbine. In this region, the wake is initially characterised by relatively compact structures, while further downstream the velocity deficit progressively decreases. As the velocity within the wake gradually approaches the undisturbed conditions, a progressive wake recovery, associated with a more homogeneous flow, can be observed.

A closer inspection of the near-wake region, shown in Figure 4.6, reveals the presence of sharp vortical structures generated by the rotor. To characterize these structures, the vorticity field is considered, where the vorticity is defined as the curl of the velocity field:

$$\boldsymbol{\omega} = \nabla \times \mathbf{u}$$

and provides a measure of the local rotation of the flow.

In the present work, the streamwise vorticity component ω_x is considered to identify the main vortical structures of the wake, where:

$$\omega_x = \frac{\partial w}{\partial y} - \frac{\partial v}{\partial z}$$

In particular, the tip vortices can be identified downstream of the blade tips, while additional vortical structures, referred to as root vortices, develop in the region close to the blade roots and the hub. These structures are initially well organised, but their interaction with the surrounding flow progressively modifies and weakens their structure as they evolve downstream. Figure 4.6 provides a detailed view of these structures through both the velocity and the vorticity fields.

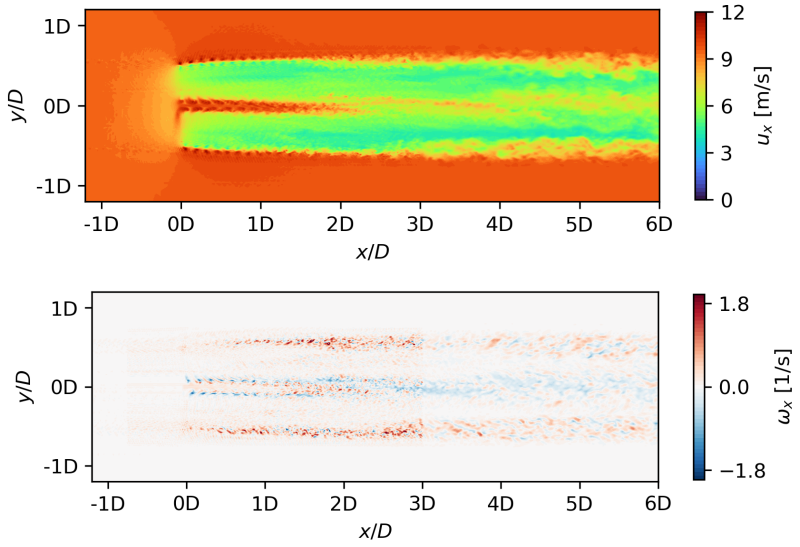


Figure 4.6: Streamwise velocity and vorticity fields near the rotor.

4.2.2 Velocity deficit, wake recovery and meandering

A direct measure of the streamwise velocity reduction with respect to the undisturbed flow allows the wake evolution to be quantitatively characterised and analysed. The velocity reduction within the wake can be quantified through the normalized streamwise velocity deficit:

$$\frac{\Delta u_x}{U_0} = 1 - \frac{u_x}{U_0} \tag{4.11}$$

where u_x is the streamwise velocity and U_0 is the undisturbed free-stream velocity.

To further characterise the spatial distribution of the velocity deficit, a Gaussian

function is fitted to the deficit profile at different downstream positions[11]. The fitted function is defined as:

$$\Delta u(y) = A \exp\left[-\frac{(y - y_c)^2}{2\sigma^2}\right] \tag{4.12}$$

where A represents the deficit amplitude, y_c is the position of the wake centre, and σ is the Gaussian width parameter, which provides a characteristic measure of the lateral extent of the wake.

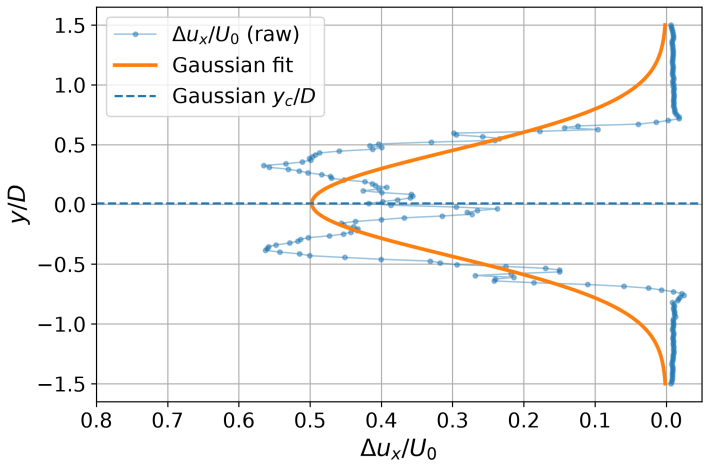


Figure 4.7: Velocity deficit profile and Gaussian fit at $t = 500$ s and $x/D = 4$.

Figure 4.7 shows a representative velocity-deficit profile at $x/D = 4$, together with its corresponding Gaussian fit. The deficit is concentrated around the wake centre and decreases towards the outer region, where the streamwise velocity progressively approaches the free-stream value. The Gaussian representation therefore provides a simplified description of the spatial distribution of the velocity deficit and allows the wake centre and wake width to be identified at different downstream positions.

The velocity-deficit profiles can be evaluated at different downstream positions to analyse the spatial evolution of the wake. Figure 4.8 shows the velocity-deficit profiles obtained at $t = 500$ s for $x/D = 4$, 8, 13.5 and 17.5. With increasing downstream distance, the wake progressively spreads and becomes wider, while the magnitude of the velocity deficit decreases as the flow recovers. The corresponding variation of the fitted wake centre position indicates that the wake does not remain aligned with the rotor axis throughout the downstream evolution.

[11]The Gaussian fit provides a representation of the velocity-deficit profile, obtained by fitting a Gaussian function to the velocity deficit across the wake. In this procedure, the fitted centre represents the wake-centre position, while the fitted width characterises the spatial extent of the wake.

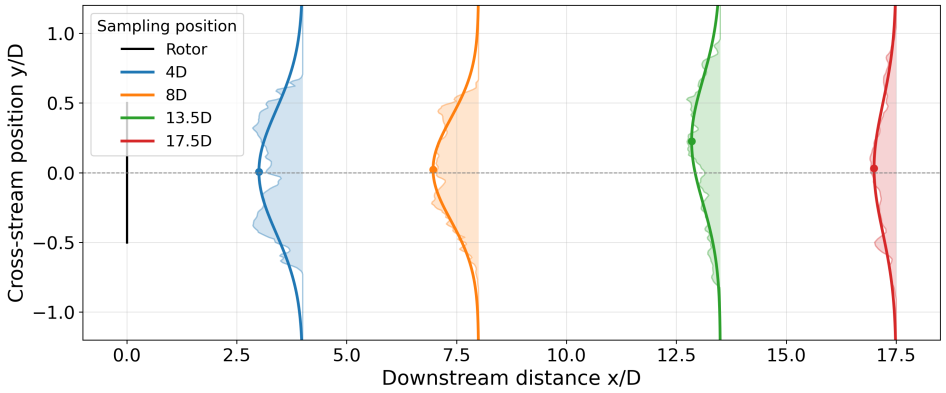


Figure 4.8: Velocity deficit profile and Gaussian fit at $t = 500$ s and at different downstream positions $x/D = 4, 8, 13.5, 17.5$.

The lateral displacement of the wake centre observed at different downstream positions can be further analysed by considering its temporal evolution. This large-scale and low-frequency behaviour, known as *wake meandering*, is associated with the interaction of the tip and root vortices with the undisturbed flow[12]. The wake meandering frequency can be quantified by tracking the position of the wake centre over time at different downstream positions. The results are displayed in Table 4.2.

x/D	f_m [Hz]	St_m [-]	T_m [s]
4.0	0.0156	0.205	64.0
8.0	0.0391	0.513	25.6
13.5	0.0156	0.206	64.0
17.5	0.0234	0.308	42.7

Table 4.2: Dominant wake meandering frequency, Strouhal number and corresponding period at different downstream positions.

4.3 Effect of surge motion on the wake

The effect of platform surge motion on the wake is now investigated by comparing the moving configurations with the static reference case. The analysis focuses on the modifications to the wake velocity field and wake dynamics induced by the streamwise motion of the rotor at different surge frequencies.

[12]In a more realistic scenario that considers the Atmospheric Boundary Layer, this phenomenon is generally more pronounced and can affect the wake behaviour.

4.3.1 Comparison of static and surge cases

For both cases, the simulation is initially continued from $t = 0\,\mathrm{s}$ to $t = 250\,\mathrm{s}$ with the platform kept fixed, while the blades rotate as in the case of the reference fixed-bottom configuration. As described in the previous section, this initial stage allows the flow to develop and reach a statistically stationary state before surge is introduced. At $t = 250\,\mathrm{s}$, the prescribed harmonic surge motion is activated and maintained. This intermediate stage allows the flow to develop and adjust to the imposed motion and provides sufficient time for the transient effects to decay. Finally, the simulations are run from $t = 600\,\mathrm{s}$ to $t = 950\,\mathrm{s}$. The latter interval is considered for the analysis of the wake under surge motions.

To assess the effect of the imposed surge motion on the turbine power output, the temporal evolution of the rotor power is analysed. In the fixed-bottom reference case, the instantaneous power exhibits small rapid oscillations around an approximately constant mean value of 3.2 MW; consequently, the corresponding mean power remains nearly constant near that value throughout the analysed time interval. Figure 4.9 shows the instantaneous and mean power of the wind turbine in the fixed-bottom configuration.

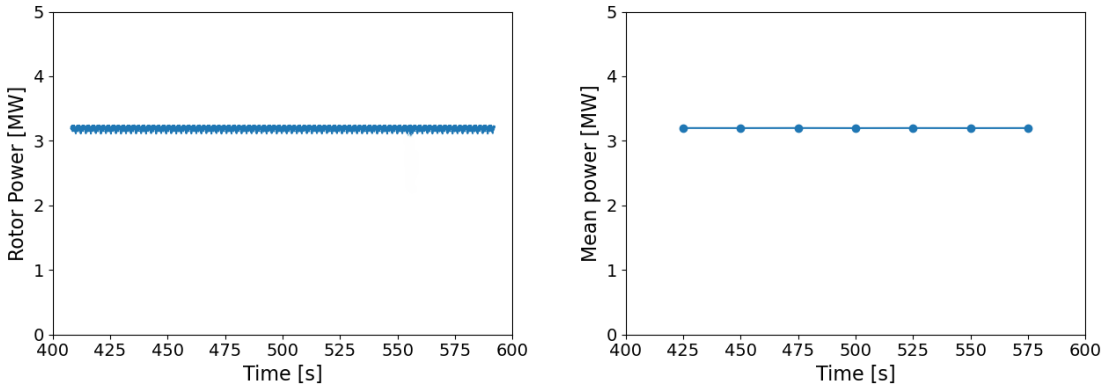


Figure 4.9: Instantaneous (left) and mean (right) rotor power in the fixed-bottom reference case.

To reduce the influence of the small and rapid power fluctuations associated with rotor rotation, the instantaneous rotor power is averaged over consecutive intervals corresponding to four rotor revolutions. The rotor period is approximately $T = 6.5\,\mathrm{s}$, resulting in an averaging window of $4T = 26\,\mathrm{s}$.

In the surge cases, the instantaneous rotor power exhibits larger oscillations than in the fixed-bottom configuration. These fluctuations are associated with the periodic streamwise platform motion, which modifies the relative inflow velocity

experienced by the rotor, and since the power is strongly dependent on the incoming velocity, even small variations in the relative velocity can produce appreciable variations in the instantaneous power. As shown in Figure 4.10, the higher frequency surge case ($St = 0.91$) exhibits more rapid power fluctuations and a larger fluctuation amplitude than the lower-frequency case ($St = 0.39$).

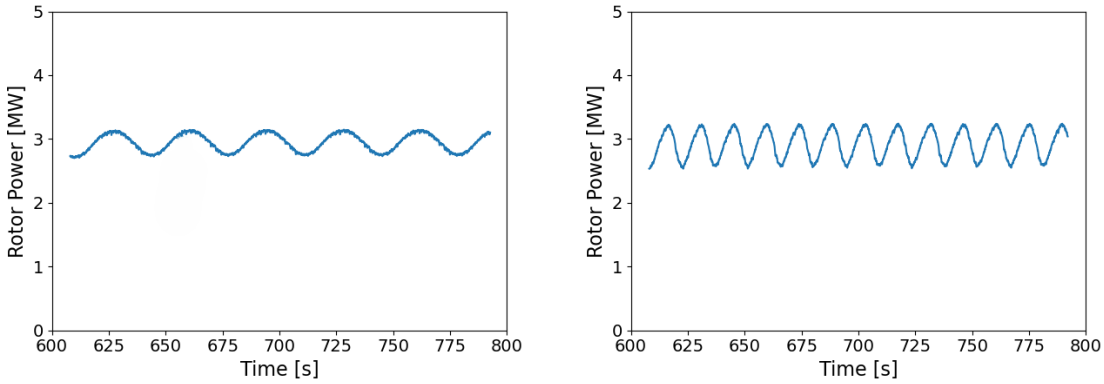
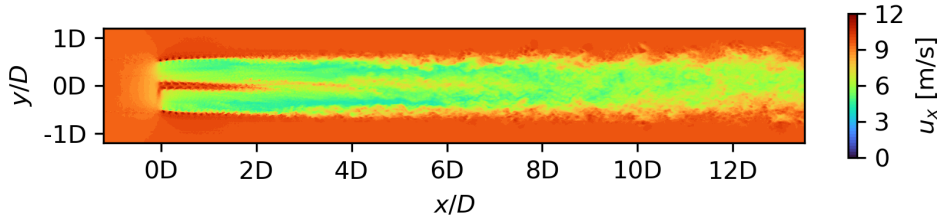


Figure 4.10: Instantaneous rotor power in the surge configuration cases (St=0.39 on the left, St=0.91 on the right).

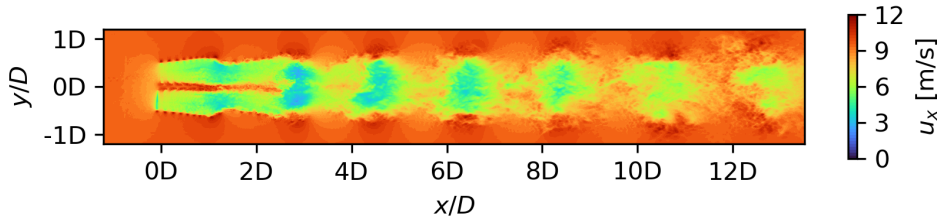
After applying block averaging to both surge cases, a small difference in the mean power can be observed: the mean rotor power for the low-frequency surge case is approximately 2.94 MW, while the mean power for the high-frequency surge case is 2.91 MW. Both values are lower than the approximately 3.2 MW obtained for the fixed-bottom configuration.

The reduction in the mean power observed in the present simulations is consistent with previous numerical studies showing that surge motion can modify the average power output of a floating wind turbine [22]. In particular, the effect of surge on the mean power depends on the operating conditions and the characteristics of the imposed motion: at low tip-speed ratios, an increase in surge frequency can lead to a decrease in the mean power, whereas the opposite trend may occur at higher tip-speed ratios. For the operating conditions considered here, the imposed surge motion results in stronger periodic fluctuations of the instantaneous power and in a reduction in the mean power of approximately 8 % and 9 % for the $St = 0.39$ and $St = 0.91$ cases, respectively, with respect to the fixed-bottom reference case.

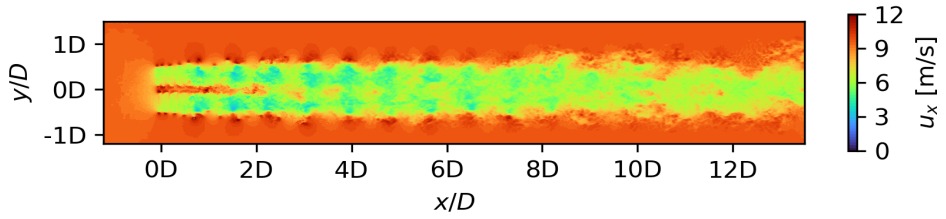
The influence of surge motion on the instantaneous wake structure is now investigated by analysing instantaneous streamwise velocity fields. This analysis allows the different flow structures generated downstream to be identified and compared.



(a) Fixed-bottom reference case



(b) $St = 0.39$



(c) $St = 0.91$

Figure 4.11: Comparison of the instantaneous streamwise velocity fields for the fixed-bottom reference case and the two surge motion cases ($St = 0.39$ and $St = 0.91$).

As shown in Figure 4.11, in the fixed-bottom configuration, the wake appears as a continuously developing turbulent flow, characterized by persistent small-scale fluctuations and vortical structures generated by the rotating blades. In contrast, the surge cases exhibit a more organized and intermittent wake evolution, with large-scale structures periodically generated and convected downstream. Similar coherent structures induced by surge motion have also been reported in previous studies [23].

The frequency of the imposed surge motion strongly affects the spatial distribution of these structures. In the low-frequency case, the wake is characterized by larger and more widely spaced ones, which are released at longer time intervals. Conversely, the high-frequency case produces a larger number of structures that are closer to each other and appear to have a smaller characteristic size. This frequency-dependent behaviour is related to the periodic variation of the rotor position and relative inflow velocity induced by the surge motion. The resulting unsteadiness modifies the wake development and may affect the downstream wake recovery.

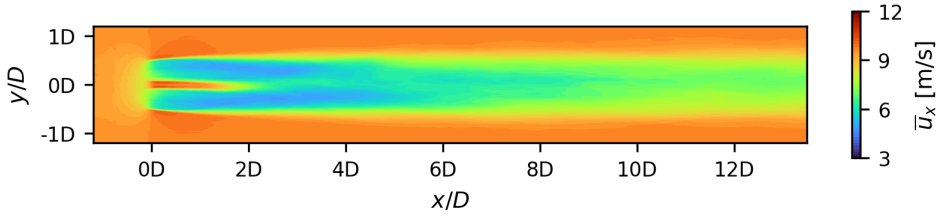
Figure 4.12 shows the mean streamwise velocity field for the three cases con-

sidered. The regions characterized by warmer colours correspond to flow velocities closer to the undisturbed inflow velocity, whereas the colder-coloured regions indicate a stronger velocity deficit within the wake. The mean streamwise velocity deficit is defined as:

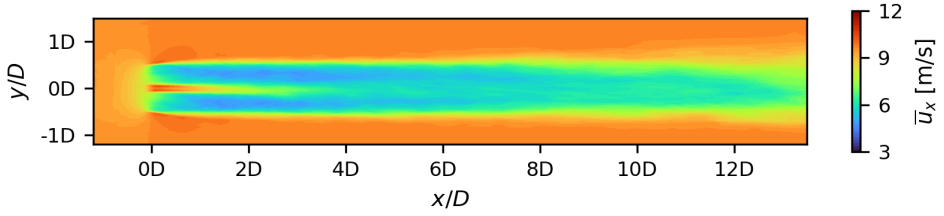
$$\frac{\Delta \overline{u}_x}{U_0} = 1 - \frac{\overline{u}_x}{U_0} \tag{4.13}$$

where $\overline{u}_x$ is the time-averaged streamwise velocity and U_0 is the undisturbed inflow velocity. This quantity measures the reduction in the mean streamwise velocity with respect to the incoming flow, while wake recovery is associated with the downstream reduction of the velocity deficit.

A qualitative comparison of the three cases shows that the low-frequency surge case ($St = 0.39$) is associated with a faster wake recovery than the high-frequency case ($St = 0.91$). In particular, the progressive recovery downstream of the rotor leads to an increase in the mean streamwise velocity and a corresponding reduction in the velocity deficit. This behaviour is more pronounced in the lower-frequency surge motion, indicating a faster wake recovery in the lower-frequency case ($St = 0.39$).



(a) $St = 0.39$



(b) $St = 0.91$

Figure 4.12: Comparison of the mean streamwise velocity fields for the two surge motion cases ($St = 0.39$ and $St = 0.91$).

Although previous investigations of floating wind turbines under surge motion considered an atmospheric boundary layer (ABL) instead of a uniform inflow adopted in the present simulations [20], these results are consistent with previous studies.

To quantitatively assess the wake recovery of wind turbines, the maximum mean velocity deficit is evaluated at different downstream positions. At $x/D = 8$, the

maximum mean velocity deficit is approximately 45% for the fixed-bottom case, 36% for the $St = 0.39$ case, and 41% for the $St = 0.91$ case. Therefore, the low-frequency case reduces the velocity deficit by approximately 20.3 % with respect to the fixed-bottom configuration, while the high-frequency case achieves a reduction of approximately 8.4 %. A similar trend can be observed at other downstream positions, as reported in Table 4.3. The differences between the three cases are more pronounced in the intermediate downstream region, particularly at $x/D = 8$ and $x/D = 10$, while further downstream the velocity deficits become more similar.

x/D	Maximum mean velocity deficit [%]		
	Fixed-bottom	$St = 0.39$	$St = 0.91$
4.0	55.8	47.6	48.7
8.0	45.3	36.1	41.5
10.0	41.3	29.7	38.7
13.5	32.2	28.0	30.8

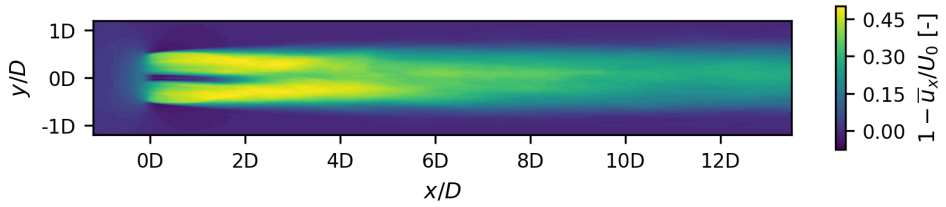
Table 4.3: Maximum mean streamwise velocity deficit across the cross-stream direction at different downstream positions for the fixed-bottom and surge-motion configurations.

The results show that the imposed surge motion positively affects wake recovery for both investigated frequencies, although the improvement is more pronounced for the lower-frequency case. Indeed, the $St = 0.39$ configuration exhibits the lowest velocity deficit and, therefore, the fastest wake recovery, as also shown in Figure 4.13.

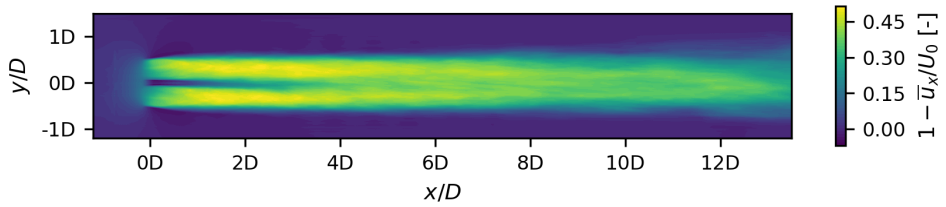
This behaviour is consistent with the results reported in previous studies of floating offshore wind turbines under harmonic surge motion [24, 20]. These studies show that the effect of surge motion on wake recovery is strongly influenced by the motion frequency: in particular, the wake recovery does not increase with the Strouhal number, but instead an intermediate range of surge frequencies may produce a more favourable wake response, while higher frequencies may result in a less effective recovery.

The physical mechanism responsible for this behaviour is related to the formation of large-scale coherent structures induced by the periodic platform motion. The harmonic motion of the platform and the variation in the relative velocity encountered by the blades during the motion lead to the generation of large-scale structures, which enhance the exchange of momentum between the undisturbed surrounding flow and the low-momentum region behind the rotor, promoting the transport of high-momentum fluid toward the wake centre and contributing to the reduction of the mean velocity deficit.

The present results suggest that a similar mechanism may occur in the investigated simulations: the lower-frequency surge case ($St = 0.39$) generates larger and more widely spaced structures compared to the higher-frequency case ($St = 0.91$). Larger structures may result in a more effective exchange of momentum toward the wake centre and, consequently, the recovery of the mean velocity field. Conversely, the smaller structures generated by the higher frequency case are more closely spaced and appear to have a smaller characteristic spatial scale: these structures may therefore contribute less effectively to downstream momentum transport, resulting in a slower wake recovery.



(a) $St = 0.39$



(b) $St = 0.91$

Figure 4.13: Comparison of the velocity deficit for the two surge motion cases ($St = 0.39$ and $St = 0.91$).

The observed improvement in wake recovery is particularly relevant for the performance of floating offshore wind farms: a faster recovery of the mean streamwise velocity reduces the velocity deficit experienced by downstream turbines, potentially increasing their available power.

This interpretation is further supported by the resolved turbulent kinetic energy (TKE) extracted from the wake sections. The turbulent kinetic energy is defined as:

$$k = \frac{1}{2}(u'^2 + v'^2 + w'^2) \tag{4.14}$$

where u', v' and w' denote the velocity fluctuations in the streamwise, lateral and vertical directions, respectively, and are defined as the difference between the instantaneous and the mean velocity components (i.e., $u' = u - \overline{u}$). By considering a sampling

band centred at the target downstream position, defined by $|x - x_{target}| \leq 0.05D$ in the streamwise direction and $|y| \leq 1.2D$ in the cross-stream direction, and by averaging the turbulent fluctuations over this region, the following equation is obtained:

$$k = \frac{1}{2}(\overline{u'^2} + \overline{v'^2} + \overline{w'^2})$$

In the present work, a normalised version of the equation written above is used, and it is found by dividing k by the square of the undisturbed velocity:

$$k^* = \frac{k}{U_0^2} \tag{4.15}$$

By comparing the three cases, it can be observed that both the lower-frequency and higher-frequency cases exhibit an increased resolved TKE compared to the fixed-bottom case. This trend is consistent with the differences observed in wake recovery: the $St = 0.39$ case is characterised by a higher resolved TKE over the intermediate region between $x/D = 4$ and $x/D = 10$, where it also exhibits a lower mean velocity deficit; at $x/D = 13.5$, the difference in TKE between the two surge cases becomes less pronounced, matching the reduced difference in their mean velocity deficits (see Table 4.3).

x/D	Normalized mean resolved TKE ($\times 10^2$) [-]		
	Fixed-bottom	$St = 0.39$	$St = 0.91$
4.0	0.141	1.028	0.662
8.0	0.333	0.956	0.511
10.0	0.392	0.839	0.513
13.5	0.428	0.703	0.653

Table 4.4: Mean resolved normalised TKE, analysis at different downstream positions for the fixed-bottom and surge-motion configurations.

Reynolds shear stresses provide additional information on the turbulent interactions that occur within the wake. The Reynolds stress tensor is defined as $\tau_{ij}^R = -\rho\overline{u_i' u_j'}$, and its matrix form is:

$$-\rho \begin{bmatrix} \overline{u'u'} & \overline{u'v'} & \overline{u'w'} \\ \overline{v'u'} & \overline{v'v'} & \overline{v'w'} \\ \overline{w'u'} & \overline{w'v'} & \overline{w'w'} \end{bmatrix}$$

where ρ is the fluid density. The off-diagonal terms represent the Reynolds shear stresses and describe the turbulent transport of momentum associated with corre-

lated velocity fluctuations; in particular, the $\overline{u'v'}$ component is associated with the turbulent transport of streamwise momentum in the lateral direction. In the present analysis, the $\overline{u'v'}$ component is considered in its normalised form:

$$\frac{\overline{u'v'}}{\overline{U_0^2}}$$

Since the spatial average of the $\overline{u'v'}$ component may be affected by the cancellation of positive and negative contributions within the band region considered, the mean absolute value[13] is used as a measure of the magnitude of the $u'v'$ correlations within the sampling region:

$$\frac{1}{N}\sum_{i=1}^{N}\frac{\left|\overline{u_i'v_i'}\right|}{\overline{U_0^2}}$$

where N is the number of sampling points considered for the average in the band region.

The mean absolute value of the normalized $\overline{u'v'}$ component is higher for the surge cases in the near-wake region, particularly for $St = 0.39$: this indicates strong correlations between streamwise and lateral velocity fluctuations induced by the platform motion, and suggests enhanced momentum exchange within the wake. However, as shown in Table 4.5, the far-wake region does not show the same behaviour: the magnitude of $u'v'$ correlations does not increase together with the wake recovery. Therefore, a larger magnitude of the normalized $u'v'$ correlations does not vary consistently with the observed wake recovery.

x/D	Mean absolute normalized $\overline{u'v'}$ component ($\times 10^3$) [-]		
	Fixed-bottom	$St = 0.39$	$St = 0.91$
4.0	0.385	1.891	1.166
8.0	0.959	1.152	0.919
10.0	1.137	0.746	0.831
13.5	1.194	1.258	2.258

Table 4.5: Mean absolute value of the normalized $\overline{u'v'}$ component at different downstream positions.

These results are consistent with the observed wake recovery: the stronger recovery in the low-frequency surge case leads to a smaller velocity deficit in the far-wake

[13]This quantity should be interpreted as an indicator of the intensity of the turbulent correlations rather than as the Reynolds shear stress itself, which would include the factor $-\rho$.

region and, consequently, to lower $\overline{u'v'}$ correlations compared with the other configurations. This suggests that, as the wake recovers and the velocity deficit decreases, the turbulent momentum transport associated with the streamwise and lateral velocity fluctuations also becomes less pronounced. The higher values observed in the other configurations may instead be associated with the persistence of a larger velocity deficit further downstream, indicating that the momentum exchange remains more active where the wake has not yet fully recovered.

4.4 Hilbert POD analysis on wind turbine wakes

Before applying the POD to the different turbine configurations, a temporal convergence analysis was initially planned to assess the sensitivity of the modal decomposition to the temporal sampling of the flow field. The objective was to identify an appropriate time interval between consecutive snapshots, avoiding unnecessary dense sampling while retaining the dominant temporal features of the wake dynamics.

The analysis was intended to be performed on the static turbine configuration using the same three-dimensional sampling domain adopted for the subsequent POD analyses. The flow field was sampled over the same total time interval, while three different snapshot intervals were considered: $\Delta t_{POD} = 0.25$, 0.5, and 1.0 s. The analysis was intended to compare the mean velocity field and the calculated fluctuating energy. A sampling interval is considered adequate if further refinements of the temporal sampling produce only minor changes in the dominant modal energy distribution.

However, the POD calculation with the finer temporal sampling could not be completed due to the large size of the resulting snapshot matrix. Moreover, the current implementation does not account for a redistribution of sampled cells among the MPI ranks after the application of the inclusion box, which can cause a non-uniform workload between ranks during the SVD computation and may contribute to the computation limitations encountered.

Therefore, the temporal sampling analysis performed in the present work considers only the larger time interval described above, resulting in a limited number of snapshots. The results presented in the next section should therefore be considered preliminary. Nevertheless, these results provide some initial indications of the behaviour of the POD decomposition and can be used as a first reference for future work. Due to the limited number of snapshots, the resulting modal structures should be interpreted with caution.

4.4.1 Hilbert POD analysis

By applying the HPOD analysis to the wind turbine simulations, it is possible to investigate the physical behaviour of the large and most energetic coherent fluctuations in the wake. For the static case, the energy distribution among the first eight HPOD modes shows that there is no a single mode that dominates the turbulent flow field: the first mode accounts for 18.26 % of the reference energy, while the first three modes together account for approximately 48% of it. The similar energy content of the extracted modes indicates that the turbulent fluctuations are distributed among different coherent structures instead of being concentrated in a single mode.

HPOD mode	Energy content (%)	Cumulative energy content (%)
1	18.26	18.26
2	15.68	33.92
3	14.31	48.23
4	12.61	60.83
5	11.88	72.71
6	10.04	82.76
7	8.74	91.50
8	8.50	100

Table 4.6: Energy content of eight HPOD modes for fixed-bottom configuration.

This result can be linked to the wake development observed in the previous section: in the static case, the TKE progressively increases in the downstream direction, while the mean velocity deficit decreases thanks to the momentum exchange within the wake.

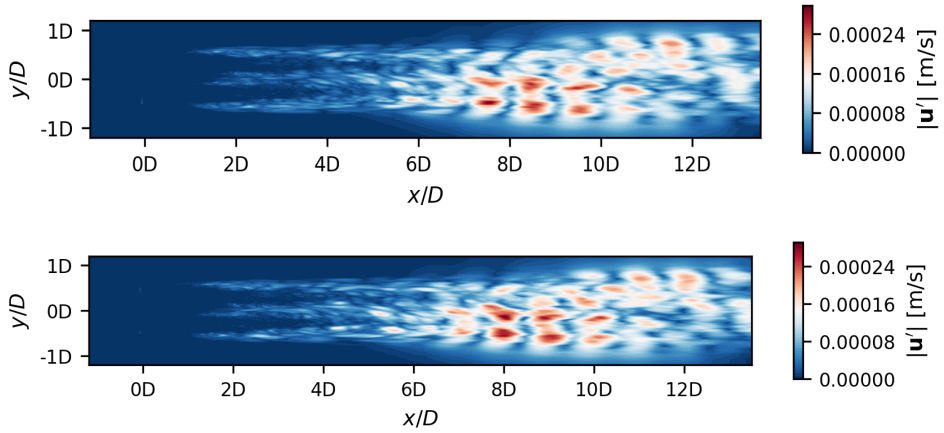


Figure 4.14: Velocity magnitude of the first POD mode: both real (top) and imaginary (bottom) parts.

Figure 4.14 shows the magnitude of both the real and imaginary components of the first HPOD mode. The magnitude represents the amplitude of the velocity fluctuations in different regions of the domain: values close to zero represent a negligible contribution of the mode, while higher values indicate regions where the mode has a larger amplitude. The most visible structures become visible at approximately $7D$ downstream of the rotor and remain particularly intense up to approximately $11D$ from the rotor. The imaginary part represents the same complex-valued mode with a phase shift and provides information about the evolution of the structures.

The second HPOD mode exhibits a different spatial organisation, with stronger structures concentrated around the blade-tip regions and extending downstream from approximately $7D$ to $11D$. The localisation of these structures suggests a connection with the coherent structures generated in the tip region.

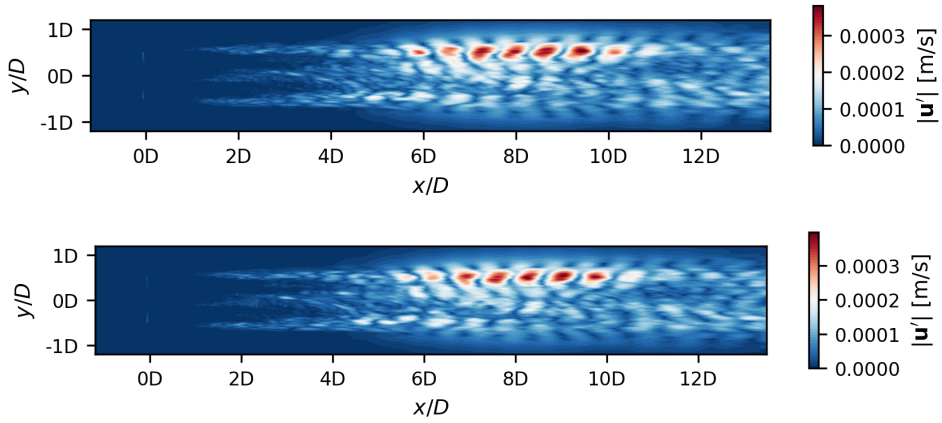


Figure 4.15: Velocity magnitude of the second POD mode: both real (top) and imaginary (bottom) parts.

The asymmetry observed in the spatial modes suggests that the current POD analysis does not adequately capture the fluctuating field and the associated wake dynamics. This aspect requires further investigations through additional analyses and simulations.

Overall, the HPOD analysis applied to the fixed-bottom case indicates that the wake is characterised by several coherent structures with comparable energy content, rather than being governed by a single dominant structure.

By considering the streamwise and transverse velocity components, u' and v', respectively, of the first HPOD mode, some consideration on the turbulent fluctuations may be made. For a given component, the red regions represent positive velocity fluctuations, while the blue regions represent negative velocity fluctuations. The analysis of the streamwise velocity component provides information on the behaviour of streamwise velocity fluctuations, while the transverse component provides

complementary information on the lateral motion. Considering the coupling of the two velocity components, it is possible to visualise how the flow fluctuations behave and their spatial organisation.

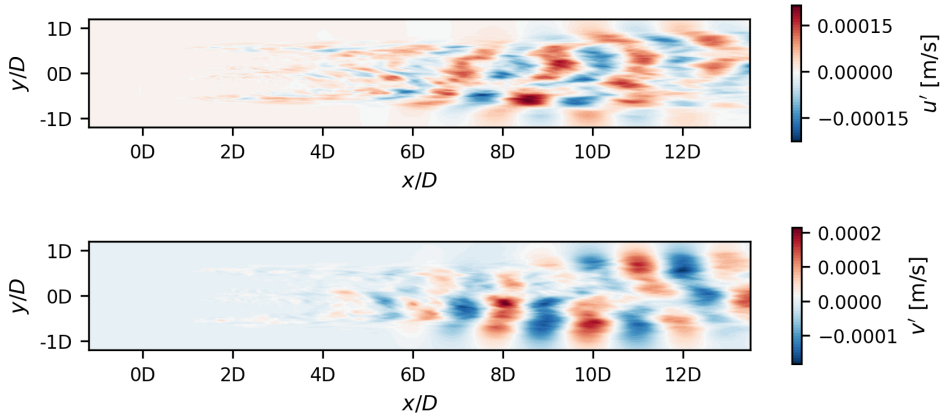


Figure 4.16: Streamwise (top) and transverse (bottom) velocity components of the first POD mode.

Applying the HPOD to the surge moving wind turbine configurations provides further insight into the wake behaviour for those cases. The low-frequency surge motion is characterized by a significantly different modal energy distribution compared to the fixed-bottom reference case. The first mode alone accounts for almost 72 % of the total fluctuating energy captured, while the first four modes together contain more than 90 % of the cumulative energy, showing that most of the fluctuation energy captured by the HPOD decomposition is concentrated within the first few modes and, in particular, the fluctuating field is dominated by one mode. The results are summarised in Table 4.7.

HPOD mode	Energy content (%)	Cumulative energy content (%)
1	71.93	71.93
2	12.50	84.43
3	3.78	88.21
4	2.71	90.92
5	2.47	93.40
6	2.30	95.70
7	3.20	97.90
8	2.10	100

Table 4.7: Energy content of eight HPOD modes in $St = 0.39$ configuration.

Figure 4.17 shows the spatial distribution of the first HPOD mode, including its magnitude and both the streamwise and transverse velocity fluctuation components,

whose spatial distribution can provide information on the spatial organisation of the coherent structures within the wake.

For example, at approximately $4D$ downstream of the rotor, the streamwise component shows a positive fluctuation in the central region, whereas the transverse component shows both positive and negative regions. This indicates that the dominant mode is characterized by a combination of streamwise and transverse velocity fluctuations, and their coupling can explain how the coherent structures develop downstream of the rotor.

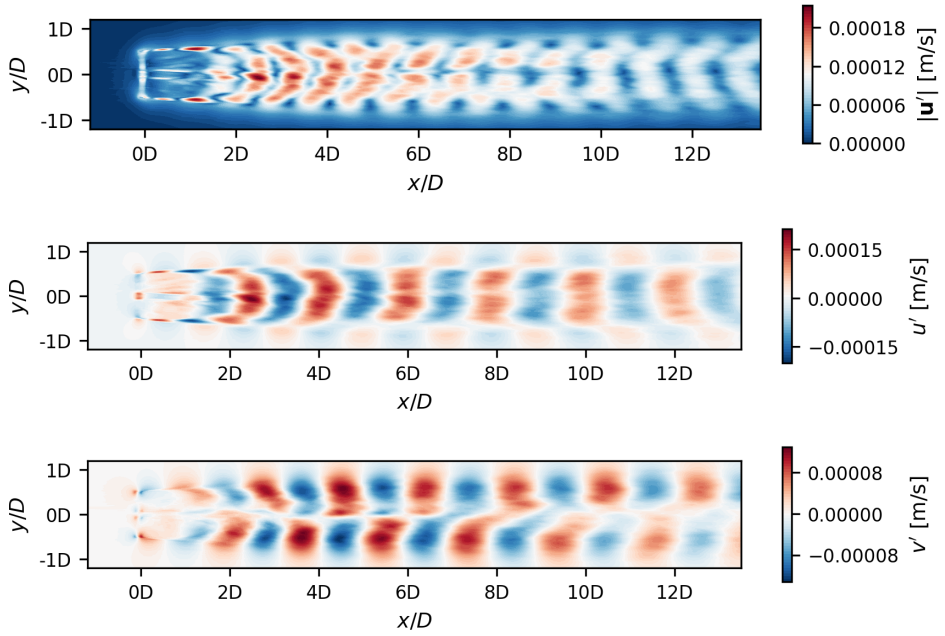


Figure 4.17: Magnitude (top), streamwise velocity component (centre), and transverse velocity component (bottom) of the first POD mode for the low-frequency surge configuration.

The alternating distribution of the streamwise and transverse velocity fluctuations is associated with a rotational behaviour of the flow field in the wake. These large structures contribute to the momentum exchange between the outer region and the wake centre region: in particular, the strong concentration of energy in the first mode suggests that this large-scale structure plays an important role in the wake recovery. It is particularly interesting to notice how the HPOD identified the dominant structure and how it is possible to link the behaviour of the fluctuating components to the results of the previous section: the most energetic structure of the fluctuating field is the one that promotes the recovery of the mean velocity in the streamwise direction. This result may explain why the lower-frequency surge case has the fastest wake recovery between the three configurations analysed.

Similar considerations can be made for the high-frequency surge case. The modal energy distribution is less concentrated in the first mode than in the low-frequency configuration, although it remains more concentrated than in the fixed-bottom reference case. In this configuration, the first mode accounts for approximately 48 % of the measured energy content, while the first six modes contain approximately 90 % of the cumulative energy, indicating that the fluctuation energy content is distributed among a larger number of modes compared with the low-frequency surge configuration. The corresponding energy distribution is reported in Table 4.8.

HPOD mode	Energy content (%)	Cumulative energy content (%)
1	48.24	48.24
2	14.98	63.22
3	10.59	73.81
4	6.03	79.84
5	5.50	85.35
6	5.41	90.76
7	4.87	95.63
8	4.37	100

Table 4.8: Energy content of eight HPOD modes in $St = 0.91$ configuration.

The different modal energy distributions highlight how the surge frequency modifies the wake evolution, with the generation of smaller and less energetic vortices. It is possible to notice again that this outcome is in line with the results shown in the previous section.

Figure 4.18 shows the magnitude, the streamwise and transverse components of the first HPOD mode. The spatial distribution observed in the high-frequency case shows similarities with those identified for the low-frequency configuration, but with more closely spaced fluctuations. Similarly to the low-frequency case, the streamwise and transverse components show regions of positive and negative values, indicating a similar combined contribution to the other surge configuration.

The high-frequency simulation shows a similar behaviour to the low-frequency one. However, the lower energy concentration in the first mode suggests a less effective momentum exchange and a consequent slower wake recovery, consistent with the results found in the previous section.

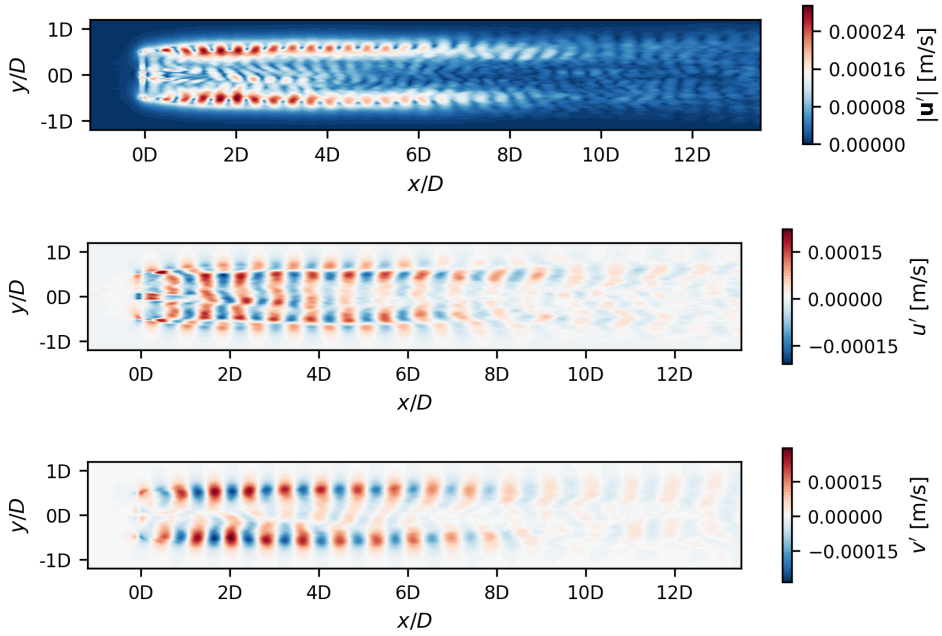


Figure 4.18: Magnitude (top), streamwise velocity component (centre), and transverse velocity component (bottom) of the first POD mode for the high-frequency surge configuration.

The influence of the surge frequency acts on both the energy concentration among the modes and the spatial organisation of the dominant coherent structures. Therefore, a deeper study on the optimal frequency range may be performed in the future to better assess this phenomenon.

Chapter 5

Conclusions

The objective of this thesis was to analyse the wake behaviour of floating offshore wind turbines and compare the wake evolution of two surge motion cases with that of a fixed-bottom configuration. The fluid dynamics simulations were performed using a Large Eddy Simulation (LES) approach coupled with the Actuator Line Method (ALM), which avoids the explicit modelling of the blades while capturing the main aerodynamic features of the flow, including the generation of tip and root vortices.

The wake characteristics were investigated by analysing the mean velocity field, the velocity deficit, turbulent kinetic energy (TKE), and Reynolds shear stresses. Particular attention was given to the influence of surge frequency on wake recovery and the associated turbulent structures.

Finally, Hilbert Proper Orthogonal Decomposition (HPOD) was applied as a tool to investigate the coherent structures developed within the wake. Its application in the three study cases provides additional information on the flow dynamics and the mechanism potentially responsible for the observed differences in wake recovery.

5.1 Summary of the work

The analyses performed on the floating offshore wind turbine configurations provide an insight into the wake dynamics under three different operating conditions. The wake evolution differs significantly between the three cases: indeed, the surge motion cases are characterised by the formation of relatively large coherent structures that detach from the blades and are subsequently convected downstream. The comparison of the mean power produced under the three different conditions shows that the fixed-bottom case achieves the highest mean power, whereas both surge motion cases exhibit a lower mean power and larger fluctuations in the instantaneous

power. However, the differences between the three configurations are not limited to the power production, but other wake characteristics were analysed to further investigate their behaviour.

The effect of surge motion on the mean velocity deficit and, consequently, on wake recovery, is positive. The low-frequency surge case exhibits the fastest wake recovery among the three configurations, followed by the high-frequency surge case. The results obtained from the velocity deficit are consistent with the TKE distribution: in particular, the low-frequency surge case presents a higher turbulent kinetic energy content in an intermediate downstream region, corresponding to the region where the wake recovery is faster. A similar behaviour can be observed for the high-frequency surge case, although with reduced intensity.

A possible physical mechanism that could explain these results is the entrainment: the increased turbulent activity induced by the surge motion may enhance the exchange of momentum between the surrounding undisturbed flow and the wake centre, leading to a faster wake recovery of the mean velocity. The larger coherent structure observed in the low-frequency case may contribute more to the momentum exchange, explaining the stronger wake recovery observed in this configuration.

To further investigate the aspects of the wake recovery and the energetic structures, the Hilbert POD was used as a supplementary tool. The HPOD analysis provides further information on the spatial organisation of coherent structures. The surge motion modifies both the energy distribution among the modes and their spatial organisation. In particular, the low-frequency case showed a strong concentration of fluctuation energy in the first mode, whose spatial structure was associated with the development of large vortical structures. This behaviour is consistent with an enhanced momentum exchange and a subsequent faster wake recovery of the mean velocity. A similar interpretation can be made for the high-frequency case, where the smaller-scale structures are characterised by a more distributed energy content among the modes and a less dominant contribution from the first mode, which is consistent with the less pronounced wake recovery.

5.2 Limitations and future work

The present work was intended as a preliminary investigation of floating offshore wind turbine wake dynamics, and several aspects can be further improved. First, the fluid dynamics simulations could be developed by simulating a more realistic atmospheric boundary layer (ABL). A more accurate representation of the incoming flow could provide a more realistic description of the wake behaviour. Moreover,

considering a wider range of surge frequencies and amplitudes would allow a better understanding of the wake response and could help to identify the most favourable conditions associated with the wake recovery.

The analysis could be extended to other platform motions, such as the pitch motion. This can be investigated both independently and in combination with the surge motion at different frequencies, in order to assess their combined effect on wake recovery. Furthermore, replacing the prescribed harmonic platform motion with a more realistic one can provide a more representative description of the aerodynamic behaviour of floating wind turbines. Finally, the analysis of an isolated wind turbine can be extended to the interactions between more wind turbines, to observe the effects on wind farms and to gain a better understanding of the efficiency of these plants.

Further improvements are also possible for the numerical implementation of the Hilbert POD framework: in particular, a more efficient distribution of computational cells among the MPI ranks could improve the efficiency of the SVD calculations. Moreover, the implementation could be optimised to handle larger datasets, improving the quality of the results.

Bibliography

[1] Yize Guo Ziwen Chen Xiaodong Wang and Shun Kang. *Numerical analysis of unsteady aerodynamic performance of floating offshore wind turbine under platform surge and pitch motions*. 2021.

[2] Xiaoxu Chen and Tengyuan Wang. "A Review of Experiment Methods, Simulation Approaches and Wake Characteristics of Floating Offshore Wind Turbines". In: *Journal of Marine Science and Engineering* (2025).

[3] Stephen B. Pope. *Turbulent Flows*. 2000.

[4] Rui Wang et al. "Discovering Symmetry Breaking in Physical Systems with Relaxed Group Convolution". In: *arXiv preprint arXiv:2310.02299* (2023).

[5] OpenFOAM documentation, Version 6, available at: https://doc.cfd.direct/openfoam/user-guide-v6/turbulence.

[6] Jens Nørkær Sørensen and Wen Zhong Shen. *Numerical Modeling of Wind Turbine Wakes*. 2002.

[7] Jens Nørkær Sørensen and Wen Zhong Shen. "Numerical Modeling of Wind Turbine Wakes". In: *Journal of Fluids Engineering* (2002).

[8] Troldborg and Niels. *Actuator Line Modeling of Wind Turbine Wakes*. 2009.

[9] L.A. Nartinez-Tossas, M.J. Churchfield, and C. Meneveau. *Optimal smoothing length scale for actuator line models of wind turbine blades based on Gaussian body force distribution*. 2017.

[10] Anindya Chatterjee. "An introduction to the proper orthogonal decomposition". In: *Science Association* (2000).

[11] Marco Raiola. "Hilbert proper orthogonal decomposition: A tool for educin advective wave packets from flow field data". In: *Institute of Fluid Mechanics* (2026).

[12] Gene H. Golub and Charles F. Van Loan. *Matrix Computation (4th ed.)* 4th ed. The Johns Hopkins University Press, 2013.

[13] Arnau Miró Benet Eiximeno. “On the Wake Dynamics of an Oscillating Cylinder via Proper Orthogonal Decomposition”. In: (2022).

[14] Lawrence Sirovich. “Turbulence and the dynamics of coherent structures. I. Coherent structures”. In: *Quarterly of Applied Mathematics* (1987).

[15] Julio Soria Shaun Davey Callum Atkinson. “Studying propagating turbulent structures in the near wake of a sphere using Hilbert proper orthogonal decomposition”. In: *Journal of Fluid Mechanics* (2026).

[16] Stefan Becker Eman Bagheri Riccardo Casali and Philipp Schlatter. “Disentangling coherent structures and the origin of swirl-switching. arXiv:2605.08849”. In: (2026).

[17] Matthew J. Churchfield Pankaj K. Jha. “Guidelines for Volume Force Distributions Within Actuator Line Modeling of Wind Turbines on Large-Eddy Simulation-Type Grids”. In: *Journal of Solar Energy Engineering* (2014).

[18] Hamid Sarlak YuanTso Li Wei Yu. “Wake Structures and Performance of Wind Turbine Rotor With Harmonic Surging Motions Under Laminar and Turbulent Inflows”. In: (2024).

[19] Peike J Messmer T Hölling M. “Enhanced recovery caused by nonlinear dynamics in the wake of a floating offshore wind turbine”. In: *Journal of Fluid Mechanics* (2024).

[20] Fernando Porté-Agel Xiao Hu. “Effect of the surge motion on wake characteristics of a floating offshore wind turbine”. In: (2026).

[21] Boris Conan Benyamin Schliffke and Sandrine Aubrun. “Floating wind turbine motion signature in the far-wake spectral content – a wind tunnel experiment”. In: *Wind Energy science* (2024).

[22] Xinliang Tian Binrong Wen and Xingjian Dong. “Influences of surge motion on the power and thrust characteristics of an offshore floating wind turbine”. In: *Energy* (2017).

[23] Wei Yu YuanTso Li and Hamid Sarlak. “Wake Structures and Performance of Wind Turbine Rotor With Harmonic Surging Motions Under Laminar and Turbulent Inflows”. In: *Wind Energy* (2024).

[24] Michael Hölling Thomas Messmer and Joachim Peinke. “Enhanced recovery caused by nonlinear dynamics in the wake of a floating offshore wind turbine”. In: *Journal of Fluid Mechanics* (2024).