Politecnico di Torino

von KARMAN INSTITUTE FOR FLUID DYNAMICS

Politecnico di Torino
Master of Science in Aerospace Engineering

Master's Thesis

Intercomparison of dynamic wake modeling frameworks for offshore wind farms

Supervisors:
Prof. Sandra Pieraccini
Dr. Wim Munters
Dr. Simone Gremmo
PhD Lorenzo Schena
PhD Benoit Foloppe

Candidate:
Domenico Ruotolo

April 2024

Abstract

Wind energy plays a crucial role in the transition towards sustainable power generation. However, the intricate flow dynamics within densely clustered offshore wind farms pose significant challenges leading to efficiency losses due to wake interactions, larger fatigue loading due to enhanced turbulence within the wind plant and aeroelastic phenomena.

The estimation of the performance of a wind plant can be done using different wake models. Analytical models are often used to predict the wake effects and to assess control-oriented strategies with a lower computation time compared to higher fidelity wake models. However, few of these engineering tools are able to perform unsteady wake simulations. There also exist medium-fidelity solvers which aim to balance the need for accurate dynamic wake modeling while keeping low computational cost.

This thesis compares various wake simulation models of different fidelities, ranging from analytical steady to multi-body unsteady, to investigate how these models relate to each other and underlying physical phenomena. The study focuses on the low-fidelity wake modeling frameworks UFLORIS and FLORIDyn, which are an extension of the steady wake framework FLORIS. The comparison also involves the medium-fidelity FAST.Farm framework which is a multi-physics engineering tool for modeling power performance and structural loads.

First, the intercomparison is carried out on a simplified wind turbine layout which consists of three turbines in a row with yaw control for the upstream turbine to assess the routine, accuracy, and flexibility of each model.

Secondly, an analysis in detail of the unsteady VKI solver, UFLORIS, is carried out and some improvements are proposed. The effects of these changes are illustrated on a simple simulation, yet retaining all physics of interest.

Finally, this solver is used to simulate a low-pressure system in the North Sea considering the full Belgian-Dutch offshore cluster. The results compare different models within UFLORIS and the real operative data of the turbines of one of the Belgian offshore wind farms. They exhibit a good agreement between the models which, however, show notable discrepancies from the real data, only capturing the general trend in the strongly dynamic condition (wind direction and amplitude change); while converging in steady conditions.

This project contributes to advancing the understanding and prediction of wake effects in offshore wind farms when dynamic features are important. Such insights are useful for optimizing wind farm layouts, enhancing energy production, and guiding the development of efficient control strategies, ultimately driving the sustainable growth of offshore wind energy.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

1.1.1 Wind energy

Wind energy has a vital role to play in tackling climate change and reducing our dependence on non-renewable resources. There is a growing expectation that the global energy system must rapidly transition from fossil fuels to carbon-free sources in line with international climate change agreements. In this context, wind energy is expected to reach a significant milestone of 1 TW of installed capacity by the middle of this year [1]. In addition, wind is expected to dominate, together with solar photovoltaic, in future scenarios, accounting for over 70% of capacity additions by 2050 [2]. While offshore wind installations currently only contribute to 15% of total wind capacity additions [2], there is undeniable potential for offshore wind to rapidly replace fossil fuels. This is due to its cost effectiveness and widespread availability as an energy technology, coupled with its ability to generate substantial amounts of power. In 2023, 21% of the new wind capacity installed in Europe was offshore. It is expected that the volume of offshore installations will continue to grow in the future, reaching one-third of the total wind energy production by 2030 [3]. Furthermore, European aspirations include the deployment of up to 450 GW of offshore wind energy across the continent's seas by 2050, with a projected installation of 212 GW in the North Sea alone [4].

One source of efficiency loss in a wind farm, both onshore and offshore, is related to wake effects. Wakes generated by wind turbines affect downstream turbines in complex and not fully understood ways, leading to a reduction in energy production, an increase in loads and a reduction in turbine lifetime [5]. From an engineering perspective, a better understanding of wake flow structure and the development of models to simulate wake effects will enable the wind industry to design wind turbine layouts and operational strategies more effectively, thereby reducing overall wake losses and driving the growth of offshore wind energy.

1.1.2 Wind turbine wake

The presence of a wind turbine in the atmospheric boundary layer influences the surrounding free stream flow both upwind and downwind. Particularly, the upwind

region affected by the wind turbine is called induction region and is characterized by a reduction in the wind speed, compared to the free stream velocity. The flow behind a turbine is called the wake and can be divided in two regions: near and far wake. Figure 1.1 offers a schematic representation of the flow regions around a wind turbine. Here we provide an overview of the main features of a wind turbine wake, which is based mostly on [6, 7].

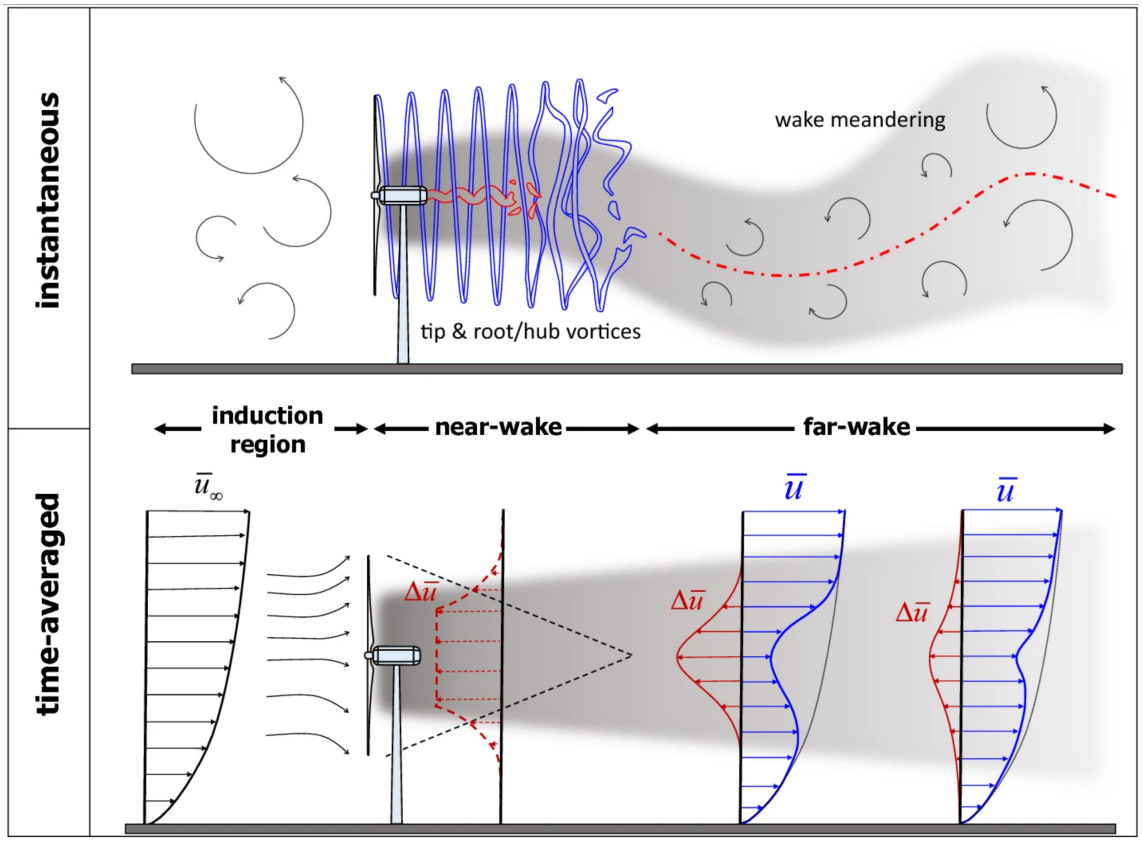


Figure 1.1: Schematic representation of the instantaneous (top) and mean (bottom) flow regions resulting from the interaction of a wind turbine with the incoming turbulent boundary layer [6].

1.1.2.1 Near wake

The near-wake region begins immediately behind the wind turbine rotor and can extend for a distance of 2-4 turbine diameters downstream. The length of this region is influenced by both the turbulence intensity and the operational settings of the wind turbine. It is characterized as three-dimensional, heterogeneous, unsteady, and directly influenced by the turbine's characteristics due to its proximity to the rotor.

During its operation, the turbine extracts momentum and energy from the flow, thus causing the presence of a positive pressure gradient in the induction region and a negative one behind the rotor with a pressure drop at the rotor plane. Despite its more complex nature, the time-averaged near wake region is typically modelled defining a potential core region where the velocity deficit is considered uniform. The region outside the potential core, called wake shear layer, is characterized by a

transition in velocity from the potential core to the flow outside the wake. Because velocity gradients exist in the shear layer, turbulence eddies are generated, turning it into a mixing region.

However, upon examining the instantaneous flow near the wind turbine, a more intricate nature of the near wake becomes evident. As illustrated in Figure 1.1, tip and root vortices with a helicoidal shape are periodically shed. These vortices result from the pressure difference between the upper and lower sides of the turbine blades. Additionally, a vortical structure at the hub height propagates downstream; this is known as the hub vortex.

1.1.2.2 Far wake

The far wake region experiences less impact from rotor-flow dynamic interactions, leading to more universal characteristics. Within this region, the streamwise velocity exhibits an axisymmetrical distribution. External flow entrainment causes the wake to expand in transversal directions as it moves downstream, and turbulence mixing accelerates the wake recovery process: the velocity deficit decreases, and small turbulent eddies dissipate more rapidly.

Considering the typical spacing of turbines in a wind farm, downstream turbines typically operate within the far wake of upstream turbines. Therefore, wake recovery becomes a key factor: the quicker the wake recovery, the closer wind turbines can operate while still generating the expected power output, allowing for narrower spacing between turbines. Wake recovery is influenced by turbulence mixing occurring due to velocity gradients in the shear layer; higher turbulence levels promote faster recovery of mean velocity in the wake.

A typical feature of the far wake is the so-called wake meandering, which consists in the dynamical transversal (in both lateral and vertical) direction of the wake centerline. These unsteady oscillations of the wake centerline are random and are induced by the large eddies with dimensions of the order of the rotor plane. Thus, wind turbines situated downstream of a meandering wake experience dynamic variations in wake characteristics which lead to yaw-misalignment. Wake meandering reduces the average velocity deficit over time but also results in increased unsteady and potentially harmful loads on downstream turbines. This phenomenon underscores the complexity and challenges in accurately assessing turbine performance and structural integrity in real-world wind farm scenarios.

1.1.3 Wake effects in a wind farm

The flow dynamics within a wind plant are more intricate than in the case of a single wind turbine wake. Integrating offshore wind farms into the electricity grid and maintaining the operational condition of wind turbines represent significant portions of the cost for offshore wind farms. To lower deployment expenses for turbines and grid infrastructure, as well as to reduce operational and maintenance costs, wind farms are often densely clustered [8]. However, this clustering approach leads to decreased overall efficiency of the wind farm due to wake losses: turbines within the farm typically produce less power and experience increased fatigue loads due to lower wind speeds and higher turbulence levels. It is estimated that wake interactions in

Figure 1.2: Photograph of the Horns Rev 2 danish offshore wind farm. Photo by: Bel Air Aviation Denmark - Helicopter Services.

offshore wind farms lead to losses of 10% in Annual Energy Production (AEP) [5]. A visualization of the complex wake interactions that can occur in an offshore wind farm is shown in Figure 1.2.

To predict the power production and the turbine loading and to design optimal control strategies for improving quality or minimizing the cost of wind energy there is a demand for numerical wake models that are reliable, efficient, and accurate.

1.1.4 Wake simulation frameworks

Simulating the physical phenomena taking place in a wind farm represents a challenge due to the wide range of spatial and temporal scales that characterise its flow dynamics. Indeed, achieving a precise representation of all the phenomena occurring is frequently challenging and demands high computational costs. Nevertheless, given that wake effects affect energy production, loads on the turbines, and efficiency, also inducing fatigue loading and aeroelastic phenomena, there is a growing demand to develop simulation tools capable of modeling the wake for both onshore and offshore wind farms.

There are several wind farm simulation tools, each characterised by a level of fidelity in modelling the wake and turbine. Here we make a distinction based on the level of accuracy in capturing the wake phenomena, identifying three main categories: low-, medium- and high-fidelity wake models; it should be stressed that there is not a unique classification of the simulators. In Table 1.1 we report an overview of some wake simulation tools commonly used. Simulators with higher accuracy generally require higher computational efforts, whereas low-fidelity tools provide a computationally inexpensive method for calculating the wind field.

Simulator	Fidelity	Model	Steady/unsteady	Computational cost
FLORIS [9]	Low	Analytical	Steady	Milliseconds
PyWake [10]	Low	Analytical	Steady	Milliseconds
FLORIDyn [11]	Low	Analytical	Unsteady	1-100 × faster than real time
UFLORIS [12]	Low	Analytical	Unsteady	1-100 × faster than real time
LongSim [13]	Low	Analytical	Unsteady	1-100 × faster than real time
DWM [14]	Medium	2D Navier-Stokes	Unsteady	1-10 × slower than real time
FAST.Farm	Medium	2D Navier-Stokes	Unsteady	1-10 × slower than real time
WFSim [15]	Medium	2D Navier-Stokes	Unsteady	1-10 × slower than real time
SOWFA [16]	High	3D Navier-Stokes	Unsteady	100 × slower than real time
AMR-Wind [17]	High	3D Navier-Stokes	Unsteady	100 × slower than real time
Xcompact3d [18]	High	3D Navier-Stokes	Unsteady	100 × slower than real time

Table 1.1: Description of some wake modeling tools.

1.1.4.1 Low-fidelity

Low-fidelity wake modeling tools are those that use analytical equations to the describe the velocity deficit behind the turbine rotor and wake recovery at low computational cost. In these models the power and thrust of the single turbine are typically computed based on look-up tables whose values are results of averaging these quantities on the rotor. Since these tools are based on parametric relations, calibration of the parameters can increase the accuracy of results [19]. However, parameter tuning is not in the scope of this work.

Many existing tools focus solely on computing time-averaged properties, thus neglecting both exogenous and internal wind farm dynamic features. Examples of steady-state wake modeling frameworks are FLORIS [9] and PyWake [10], which both include different engineering wake models to predict the power production of a wind farm while considering the wake losses for a given layout configuration at prescribed turbine settings. Despite their limited accuracy, these tools are highly valuable as they can simulate entire wind farm layouts within milliseconds, making them suitable for AEP calculations, control development [20, 21] and layout optimization [22] strategies. A detailed description of the FLORIS framework is given in Section 2.1.

However, there are also engineering dynamic models capable of capturing relevant unsteady wind farm features while maintaining low computational costs. Few tools are able to execute dynamic wake simulation tens to hundreds of times faster than real time. Examples include FLORIDyn [11], an unsteady extension of the Gaussian FLORIS model [23]; UFLORIS [12], which is an extension of the FLORIS [9] framework and thus allow to use the same engineering wake models included in it; LongSim [13], which makes use of fast engineering wake models embedded in an ambient flow field. These low-fidelity dynamic wake modeling tools are ideal for wind farm online control strategies, ensuring timely updates on a scale of seconds. Two of these dynamic tools are described in this thesis, namely FLORIDyn (Section 2.2) and UFLORIS (Section 2.3).

1.1.4.2 Medium-fidelity

Medium-fidelity tools provide a balance between accuracy in wake modeling and computational cost, offering higher accuracy than engineering models while still being less computationally expensive than high-fidelity tools. An example is given by the Wake Farm Simulation (WFSim) tool [15], a control-oriented dynamical model, which computes the wind flow at hub height solving the 2D Navier-Stokes (NS) equations discretized in time and space. The Dynamic Wake Meandering (DWM) [14] model uses small-scale turbulence filters, thus considering the wake meandering induced only by the large eddies, and capture the turbine response using aeroelastic turbine models. Another aeroelastic wind farm simulator, which builds upon the DWM model, is FAST.Farm [24]. Among these tools, FAST.Farm shows substantial promise by facilitating the simulation of various scenarios. This encompasses evaluating how individual turbines react structurally to unsteady inflows or wind transients due to time-varying turbine settings. A general description of this tool is given in Section 2.4.

1.1.4.3 High-fidelity

Finally, there exist high-fidelity wake simulation tools, which are Computational Fluid Dynamics (CFD) models typically characterized by high computational costs, often necessitating clusters to execute simulations within a reasonable timeframe. In this category, models for solving the Reynolds-averaged Navier-Stokes (RANS) equations could be included. These models are steady, meaning they do not provide a time history of the flow; however, the computational resources needed for RANS are relatively modest compared to more advanced CFD methods. Since the focus of this work does not involve high-fidelity steady solvers, further details are not provided here; interested readers are directed to [7] for examples of these models.

When talking about high-fidelity wake simulations tools, one typically refers to Large Eddy Simulation (LES) models. These models numerically solve the 3D Navier-Stokes (NS) equations by neglecting small eddies via low-pass filtering and employing sub-grid models to capture the effects of these small-scale vortices. The LES model are often used to validate lower fidelity tools [11, 25]. The Simulator fOr Wind Farm Applications (SOWFA) tool [16] is a commonly used LES solver implemented within the OpenFOAM framework; it is a finite-volume code with second-order numerical schemes in both spatial and temporal domains. Other examples of high-fidelity wake simulators are AMR-Wind [17], which perform LES for atmospheric boundary layer flows and is often used to simulate the background flow when coupled with a near-body solver; Xcompact3d [18], which is Fortran-based framework of high-order finite-difference flow solvers for LES.

1.2 Overview of the thesis

1.2.1 Goals of the project

In the previous section, three different fidelity classes of wake simulation tools were discussed, highlighting the state of the art for both steady and unsteady solvers.

Based on this overview, it is evident that each class has its own advantages and disadvantages. This project seeks to advance the understanding and prediction of wake effects in offshore wind farms, particularly focusing on dynamic features. These insights could be valuable for layout optimization, energy production enhancements, and control strategies development. Ultimately, this study could contribute to the sustainable growth of offshore wind energy. The goals of this thesis are the following.

1. The primary goal is to investigate how low- and medium-fidelity dynamic wake modeling tools can effectively capture unsteady wake features while maintaining a low computational cost. To achieve this objective, a comparison of different solvers is conducted to highlight differences in terms of runtime and accuracy for a simple wind farm configuration.

2. The secondary goal is to explore the capabilities and limitations of the dynamic engineering wake modeling framework UFLORIS, developed at the von Karman Institute for Fluid Dynamics (VKI) where this project was conducted. The objective is to address issues within the source code of the solver and propose modifications that could enhance simulations for both simple idealized and real full offshore wind farms.

1.2.2 Thesis outline

The structure of this thesis is outlined as follows.

Chapter 2 provides an overview of the wake modeling frameworks utilized in this project: FLORIS, including its steady engineering wake models; FLORIDyn and FLORIDYn, both low-fidelity dynamic wake modeling frameworks; and FAST.Farm, the sole medium-fidelity solver considered.

Chapter 3 describes the set up of three test cases. The initial section conducts a comparison of the wake modeling solvers discussed in this chapter. This comparison is based on a simple layout with three turbines in a row, along with a time-varying upstream turbine yaw control, aiming to demonstrate the capabilities of the low-fidelity solvers in capturing unsteady wake features in contrast to their steady counterparts. Subsequently, the results are compared with those obtained using FAST.Farm. The second section introduces a simplified test case designed for a comprehensive analysis of UFLORIS, detailing the modifications and enhancements made in the code. Finally, a real case study is described wherein UFLORIS is applied to simulate the full Belgian-Dutch cluster in the North Sea.

Chapter 4 presents the simulation outcomes from the preceding chapter. This chapter is divided into three sections corresponding to the three test cases. Detailed result analyses are provided along with recommendations for potential future work.

The conclusions, addressing the project's goals outlined in this chapter, are presented in Chapter 5.

Chapter 2

Wake modeling frameworks

2.1 FLORIS

The first wake modeling software used in this thesis is FLOw Redirection and Induction in Steady State (FLORIS) [9], which is an open-source control-focused wind plant simulation tool developed by National Renewable Energy Laboratory (NREL) and Delft University of Technology with support from the U.S. Department of Energy Wind Energy Technologies Office.

Within the FLORIS framework, various low-fidelity steady-state wake models are incorporated. These models enable the prediction of wind farm performance by accounting for wake effects while keeping computational costs low; however, it's important to note that FLORIS neglects dynamical effects in its simulations. The models within FLORIS are commonly constructed as a blend of velocity deficit and wake deflection models, and some also incorporate custom turbulence and combination models. Thus, in general, a wake model is composed of four submodels: the velocity deficit model, responsible for describing the wake velocity profile for a single turbine wake; the deflection model to compute the lateral displacement and the change in the wake shape due to yaw misalignment; turbulence model, representing the added-turbulence effects introduced in the flow field by operation of a wind turbine; wake combination model to account for merging of the wakes in the flow domain.

The following provides an overview of the most common wake models currently available in FLORIS: Jensen, Gauss-Curl-Hybrid (GCH) and Cumulative-Curl (CC). In particular, more details are provided on the GCH model, as it is the only one effectively utilized in this thesis work. This choice stems from the model's in-depth validation and demonstrated robustness other than the necessity to make a fair comparison with the wake model implemented in the solvers presented in the next sections. Finally, we discuss the different wake combination models available.

2.1.1 Jensen

The Jensen model [27] is one of the oldest velocity deficit models and it is still widely used due to its simplicity and relatively good accuracy. It was developed in 1983 by N.O. Jensen, who derived the model from the conservation of momentum, thus

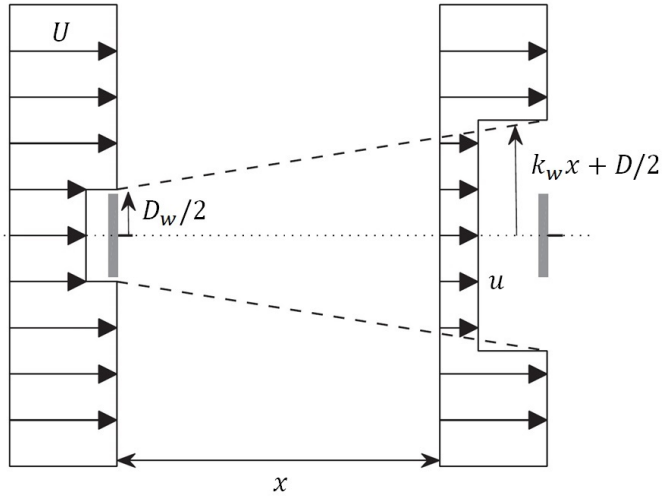


Figure 2.1: Schematic representation of the Jensen velocity deficit model. Figure adapted from [26].

adding physics to the model. The two main assumptions of the model are: the wake has a conical shape expanding linearly with increasing downstream distance from the turbine rotor and the wind speed is homogeneous in each wake plane. Thus, downstream of the originating wind turbine, the wake cone diameter D_w develop with a constant expansion coefficient, k_w, following the expression

$$D_w = D + 2k_w x \tag{2.1}$$

where D is the rotor diameter and x indicates the axial distance from the turbine.

Based on the second assumption, it follows that the wake velocity deficit exhibits a top-hat distribution: the velocity profile at each wake plane is constant and it abruptly adapts to the free stream speed outside the wake boundaries. The analytical expression for the velocity deficit is

$$U - u = U(1 - \sqrt{1 - C_T}) \left(\frac{D}{D_w} \right)^2 \tag{2.2}$$

where U is the free stream velocity, u is the wind speed at distance x from the turbine rotor and C_T is the thrust coefficient evaluated with the velocity at the rotor. Figure 2.1 shows a schematic visualization of the top-hat velocity distribution in the conical wake.

The wake expansion coefficient is the only empirical parameter governing the wake development and thus it must be selected taking into account the characteristic of the inflow to be simulated. A value which lies in the range of $0.02 - 0.05$ assures good agreement with the typical atmospheric stability and turbulence conditions for offshore applications. Thus, for a fixed value, no change in the wake field is expected for different turbulence intensity values. Moreover, since FLORIS doesn't allow the selection of different k_w values for each turbine, the Jensen model does not account for the wake sensitivity to turbulence development. Since the velocity

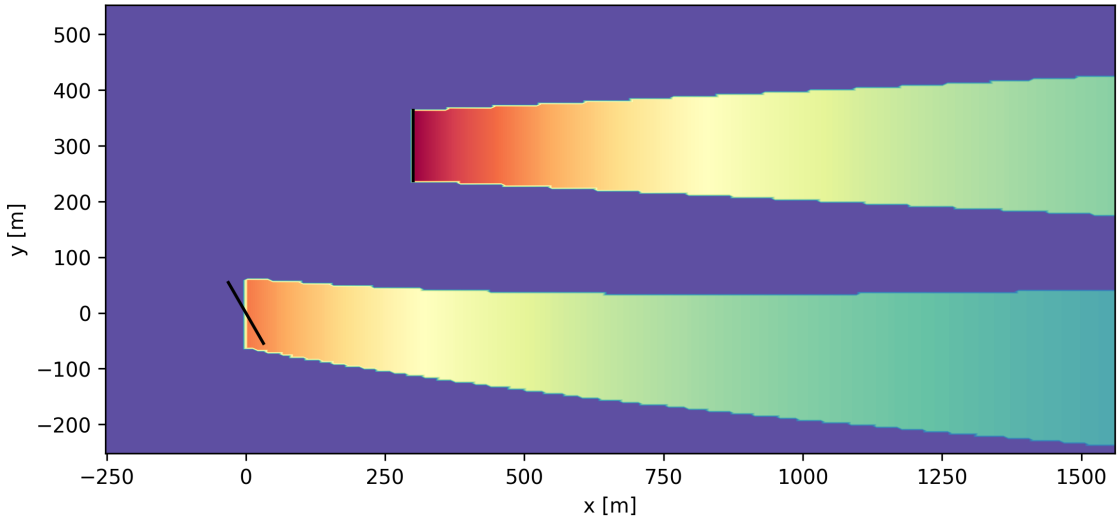


Figure 2.2: Top view of the velocity field in the Jensen-Jimenez model for both an aligned and yaw-misaligned wind turbine when the wind blows from the west. The plot, obtained in the FLORIS framework, is only intended to give a schematic view of the wake shape.

deficit is inversely proportional to the square of the wake diameter, it is clear that a higher value of the wake expansion parameter results in greater wake recovery. Therefore, a higher wake expansion parameter typically translates to increased power output for downstream turbines. Additionally, as higher turbulence levels also lead to faster wake recovery, and since turbulence intensity cannot be directly input into the Jensen model, the selection of k_w should consider the effect of turbulence as well. Generally, greater values of turbulence intensity are associated with higher k_w. In Peña, Réthoré, and Laan [26], the following relationship is suggested

$$k_w = 0.4I \tag{2.3}$$

where I is the turbulence intensity.

Regarding the wake deflection due to yaw misalignment, the Jensen model is typically associated with the Jimenez model [28]. The skew angle, which is the angle with respect to the free stream velocity, is defined as

$$\Theta = \left(\frac{D}{D_w}\right)^2 \cos^2\gamma \, sin\gamma \, \frac{C_T}{2} \tag{2.4}$$

where γ is the turbine yaw angle relative to the incident wind direction. A visualization of the application of the Jimenez deflection model to the Jensen model is provided in Figure 2.3.

2.1.2 GCH

The Gaussian-Curl-Hibryd (GCH) wake model is implemented based on the Gaussian model by Bastankhah and Porté-Agel [29] and Niayifar and Porté-Agel [30], incorporating some approximations of the curl model by Martínez-Tossas et al. [31].

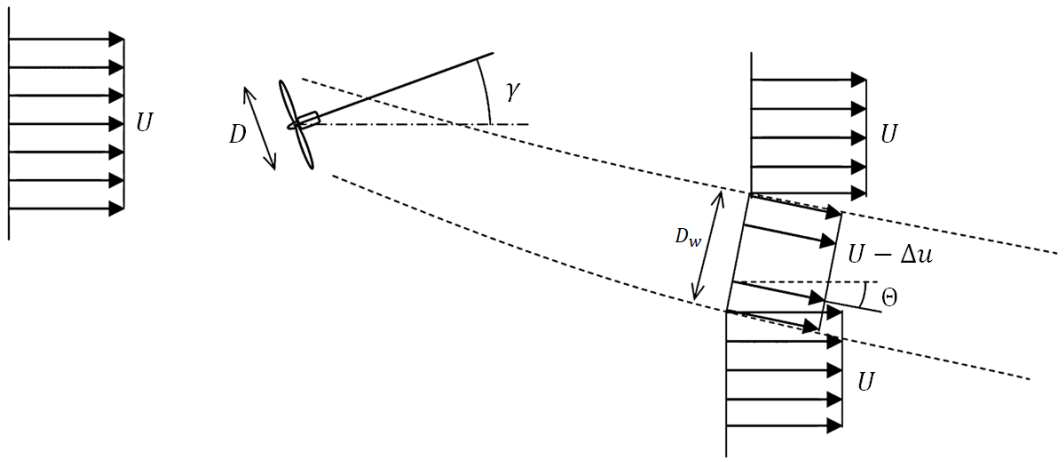


Figure 2.3: Schematic showing of the Jensen-Jimenez model. Figure adapted from [28].

2.1.2.1 Velocity deficit and turbulence models

The Gaussian wake model determines the streamwise velocity deficit at any location within a turbine's wake by employing analytical formulations derived from Reynolds-averaged Navier–Stokes (RANS) equations. The wake model is established on the principles of self-similarity theory for free shear flows, thus the Gaussian shape distribution remains constant along the streamwise direction. In this model there are a near wake zone and a far wake zone which assume different formulations for wake recovery. In the near wake there is the potential core, a conical region where both velocity and flow angle are constant. The analytical expression for the wake velocity deficit in yawed condition is the following:

$$U - u = U\left(1 - \sqrt{1 - \frac{C_T \cos\gamma}{8(\sigma_y\sigma_z)/D^2}}\right) e^{-\frac{(y-y_j-\delta)^2}{2\sigma_y^2} - \frac{(z-z_h)^2}{2\sigma_z^2}} \tag{2.5}$$

Here, δ is the wake deflection, y_j the turbine rotor center location in spanwise direction, z_h the turbine hub height, γ the turbine yaw angle; σ_y and σ_z are the wake width in y and z directions. The different values for the standard deviation of the Gaussian velocity distribution is due to the fact that the wake develops at different expansion rates [23]. These are defined as

$$\frac{\sigma_z}{D} = k_z\frac{x - x_0}{D} + \frac{\sigma_{z0}}{D} \quad \text{where} \quad \frac{\sigma_{z0}}{D} = \frac{1}{2}\sqrt{\frac{u_r}{U + u_0}} \tag{2.6}$$

$$\frac{\sigma_y}{D} = k_y\frac{x - x_0}{D} + \frac{\sigma_{y0}}{D} \quad \text{where} \quad \frac{\sigma_{y0}}{D} = \frac{\sigma_{z0}}{D}\cos\gamma \tag{2.7}$$

In the latter equations, u_r refers to the velocity at the rotor location, u_0 is the velocity in the potential core and k_y and k_z define the expansion rate respectively in lateral and vertical directions. The expansion rates depends on the turbulence intensity and on two other parameters, k_a and k_b:

$$k_y = k_z = k_a I + k_b \tag{2.8}$$

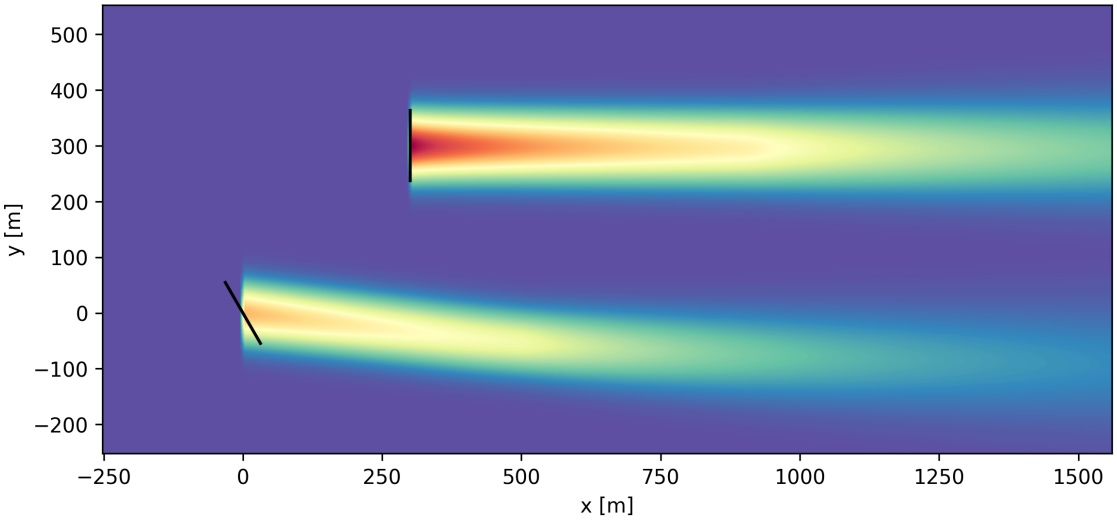


Figure 2.4: Top view of the velocity field in GCH model for both an aligned and yaw-misaligned wind turbine when the wind blows from the west. The plot, obtained in the FLORIS framework, is only intended to give a schematic view of the wake shape.

The default parameter values in FLORIS are $k_a = 0.38$ and $k_b = 0.004$.

Indicating with I_0 the initial ambient turbulence intensity (one input in the FLORIS framework) and with $N_{T,up}$ the number of upstream turbines in the plant that add turbulence to I_0 at a downstream turbine location, the turbulence intensity is defined as

$$I = \sqrt{\sum_{j=0}^{N_{T,up}} (I_j^+)^2 + I_0^2} \tag{2.9}$$

Notice that in FLORIS all turbines within a $15D$ distance give a non-zero contribute in 2.9. Based on the work of [32], the following formula is used to calculate the added turbulence intensity:

$$I_j^+ = k_{f,a} a_j^{k_{f,b}} I_0^{k_{f,c}} (x/D_j)^{k_{f,d}} \tag{2.10}$$

where D_j the rotor diameter of turbine j; $k_{f,a}$, $k_{f,b}$, $k_{f,c}$ and $k_{f,d}$ are parameters of the model that weight the foreign turbulence influence; a_j is the axial induction factor of turbine j, which is calculated based on the actuator disc theory as follows:

$$a = \frac{1}{2}(1 - \sqrt{1 - C_T}) \tag{2.11}$$

where the subscript j has been omitted for simplicity. The axial induction factor is defined as the ratio between the velocity deficit behind the turbine rotor and the undisturbed wind speed; it represents the reduction in wind speed experienced by the flow passing through the rotor disc.

2.1.2.2 Wake deflection model

The wake deflection model used to account for wake steering applications is the one based on the Gaussian deflection derived by Bastankhah and Porté-Agel [23].

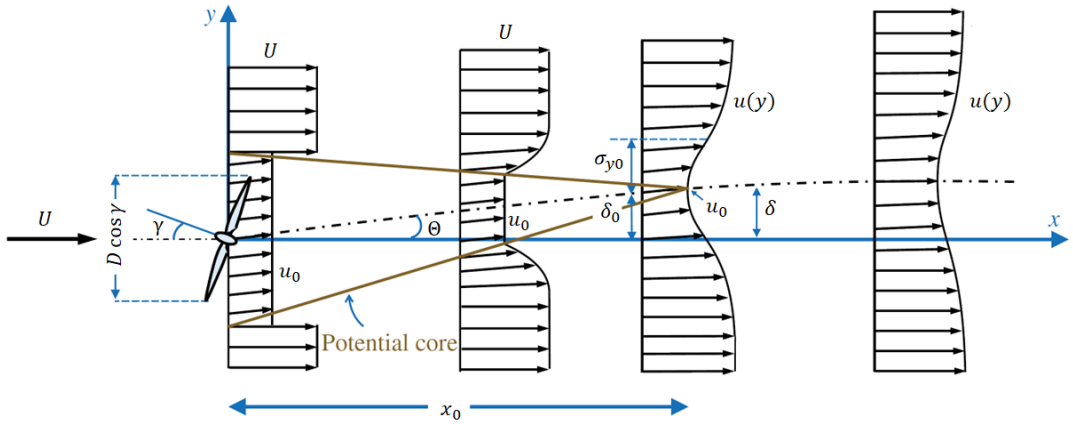


Figure 2.5: Schematic of the Gaussian wake of a yawed turbine. Figure adapted from [23].

The deflection angle at the rotor location, due to yaw error, is approximated with the following equation:

$$\Theta = \frac{0.3\gamma}{\cos\gamma}(1 - \sqrt{1 - C_T \cos\gamma}) \tag{2.12}$$

The wake deflection for near wake is then defined as

$$\delta_0 = x_0 \tan\Theta \tag{2.13}$$

where x_0 indicated the start of the far wake, i.e. the near wake length, and is computed as

$$\frac{x_0}{D} = \frac{(1 + \sqrt{1 - C_T})\cos\gamma}{\sqrt{2}[4\alpha I + 2\beta(1 - \sqrt{1 - C_T})]} \tag{2.14}$$

where α and β are two parameters that can be set in the FLORIS framework; typical values in the GCH model are $\alpha = 0.58$ and $\beta = 0.077$.

The deflection in the far wake region is then defined as

$$\delta = \delta_0 + \frac{\gamma E_0}{5.2}\sqrt{\frac{\sigma_{y0}\sigma_{z0}}{k_y k_z C_T}} \ln\left[\frac{(1.6 + \sqrt{C_T})\left(1.6\sqrt{\frac{\sigma_y\sigma_z}{\sigma_{y0}\sigma_{z0}}} - \sqrt{C_T}\right)}{(1.6 - \sqrt{C_T})\left(1.6\sqrt{\frac{\sigma_y\sigma_z}{\sigma_{y0}\sigma_{z0}}} + \sqrt{C_T}\right)}\right] \tag{2.15}$$

where $E_0 = C_0^2 - 3e^{1/12}C_0 + 3e^{1/3}$ and $C_0 = 1 - u_0/U$.

A visualization of the Gaussian wake shape of a yawed turbine is offered in Figure 2.5, where the variables are indicated for both the near and far wake.

2.1.2.3 Secondary wake effects

Compared to the original Gaussian wake model, the GCH introduces significant modifications while retaining many advantages of the underlying model. Specifically, vortices are introduced into the wake using the approximations of the curl model presented by Martínez-Tossas et al. [31]. These vortices are crucial for accounting for three phenomena:

1. they drive a yaw-based wake recovery process, leading to increased power in downstream turbines under yaw misalignment conditions of upstream turbines;

2. the interaction of counter-rotating vortices with the atmospheric boundary layer induces wake asymmetry;

3. vortices generated in wake steering are transported over long distances, influencing downstream turbines and causing the so-called secondary steering effect.

Details of these secondary wake effects introduced in the GCH model can be found in the work of [33]. Here, only the main principles are given.

To include the aerodynamics of the curled wake it is necessary to compute in the model also the transversal wake velocities, which are induced by two types of vortices: wake rotation and counter-rotating vortices. The first one is caused by the wake rotation which needs to be included since it naturally drives the wake in a preferred direction due to the rotation imposed. The wake rotation vortex strength is defined as

$$\Gamma_{wr} = \frac{\pi(a - a^2)UD}{\lambda} \tag{2.16}$$

where λ is the tip-speed ratio (or TSR) which is a settable parameter in the FLORIS framework. TSR is defined as the ratio of the tangential velocity at blade tip and the actual wind speed.

The spanwise and vertical wake velocities induced by wake rotation are defines as

$$V_{wr} = \frac{\Gamma_{wr}(z - z_h)}{2\pi[(y - y_j)^2 + (z - z_h)^2]} \left(1 - e^{\frac{-(y-y_j)^2-(z-z_h)^2}{\varepsilon^2}}\right) \tag{2.17}$$

$$W_{wr} = \frac{-\Gamma_{wr}(y - y_j)}{2\pi[(y - y_j)^2 + (z - z_h)^2]} \left(1 - e^{\frac{-(y-y_j)^2-(z-z_h)^2}{\varepsilon^2}}\right) \tag{2.18}$$

where $\varepsilon = 0.2D$, similar to [31].

In addition to the wake rotation, the model also includes counter-rotating vortices. When a turbine operates in wake steering conditions, it sheds a series of counter-rotating vortices at the rotor. In the GCH model, this collection of vortices is simply represented by a pair of two large vortices: one at the top and one at the bottom.

The circulation strength for these vortices is defined as

$$\Gamma = \frac{\pi}{8}\rho D U C_T \sin\gamma \cos^2\gamma \tag{2.19}$$

where ρ is the air density. Γ_{top} and Γ_{bottom} are computed considering the velocities at top and bottom of the rotor using the equation written above. Thus, the transversal wake velocities due to the counter-rotating vortices are defined as

$$V_{top} = \frac{\Gamma_{top}(z - z_h + D/2)}{2\pi[(y - y_j)^2 + (z - z_h - D/2)^2]} \left(1 - e^{\frac{-(y-y_j)^2-(z-z_h-D/2)^2}{\varepsilon^2}}\right) \tag{2.20}$$

$$V_{bottom} = \frac{\Gamma_{bottom}(z - z_h - D/2)}{2\pi[(y - y_j)^2 + (z - z_h + D/2)^2]} \left(1 - e^{\frac{-(y-y_j)^2 - (z-z_h+D/2)^2}{\varepsilon^2}}\right) \tag{2.21}$$

$$W_{top} = \frac{-\Gamma_{top}(y - y_j)}{2\pi[(y - y_j)^2 + (z - z_h - D/2)^2]} \left(1 - e^{\frac{-(y-y_j)^2 - (z-z_h-D/2)^2}{\varepsilon^2}}\right) \tag{2.22}$$

$$W_{bottom} = \frac{-\Gamma_{bottom}(y - y_j)}{2\pi[(y - y_j)^2 + (z - z_h + D/2)^2]} \left(1 - e^{\frac{-(y-y_j)^2 - (z-z_h+D/2)^2}{\varepsilon^2}}\right) \tag{2.23}$$

To account for the ground effect, a mirrored dummy turbine is placed above the ground in the model, thus including mirrored vortices as described in Martínez-Tossas et al. [31].

The total spanwise and vertical wake velocities are defined as

$$V_w = V_{top} + V_{bottom} + V_{wr} + V_{top,g} + V_{bottom,g} + V_{wr,g} \tag{2.24}$$

$$W_w = W_{top} + W_{bottom} + W_{wr} + W_{top,g} + W_{bottom,g} + W_{wr,g} \tag{2.25}$$

where the subscript "g" refers to the ground effect velocity component.

Finally, to account for the dissipation rate of the vortices as they move downstream, the formulation for the transversal wake velocities is

$$V = V_w \left[\frac{\varepsilon^2}{4\nu_T \frac{(x-x_j)}{U} + \varepsilon^2}\right] \tag{2.26}$$

$$W = W_w \left[\frac{\varepsilon^2}{4\nu_T \frac{(x-x_j)}{U} + \varepsilon^2}\right] \tag{2.27}$$

where ν_T is the turbulent viscosity and is computed using a mixing length model as described in King et al. [33].

The definition of transversal wake velocities in the GCH model enables the incorporation of yaw-added wake recovery: enhanced wake recovery occurs in yaw misalignment conditions because of the large-scale flow entrainment into the turbine wake field. The extra wake recovery is calculated in the GCH model by defining an added turbulence mixing term I_{mixed} induced by the previously mentioned counter-rotating vortices. For more information on how I_{mixed} is calculated, please refer to [33].

The previously discussed vortices propagate in the field and directly influence turbines located downstream of the generating turbine in a phenomenon known as secondary steering. Specifically, the transversal wake velocities generated by these vortices act similarly to an effective yaw angle, affecting the shape of wakes downstream as if the downstream turbines were also implementing wake steering, even under aligned conditions. In the GCH model this secondary steering effect is included defining the effective yaw angle $\gamma_{effective}$, which is computed as the difference

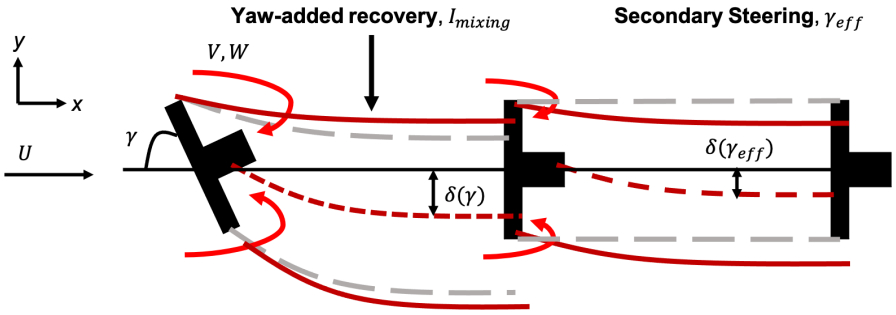


Figure 2.6: Visualization of the secondary wake effects. The expected deflection from the Gaussian model is depicted in grey, while the deflection model based on yaw-added recovery and secondary steering effects, as included in the GCH model, is shown in red [33].

between the velocity direction (based on the calculation of transversal components) and the downstream turbine direction. Thus, indicating with γ_{turb} the yaw error actually applied by the downstream turbine, the total yaw angle used to compute the wake deflection is

$$\gamma = \gamma_{turb} + \gamma_{effective} \tag{2.28}$$

Figure 2.6 provides a visualization of the changes in the deflection model resulting from the inclusion of secondary wake effects. The wake predicted by the Gaussian model is depicted in grey, while the wake predicted by the GCH model is shown in red. The spanwise and vertical velocities induced by the yaw misalignment of the upstream turbine contribute to an added recovery effect, leading to improved wake recovery with a larger wake expansion for the upstream turbine. Additionally, in the GCH model, although the downstream turbine is not actively implementing wake steering, its wake is deflected based on the effective yaw angle; instead, in the Gaussian model the downstream turbine wake centerline is perfectly aligned with the axis of the turbine rotor.

2.1.3 CC

The Cumulative-Curl (CC) model [34] is an extension of the GCH model that incorporates enhancements aimed at improving the accuracy of simulations for large wind farms. Specifically, the GCH model tends to exhibit rapid wake recovery, leading to underestimated wake losses. The CC model addresses this issue by introducing modifications that better capture wake dynamics, thus enhancing the overall predictive capability for large-scale wind energy systems.

Firstly, the CC model replace the SOSFS combination method (described in the following subsection) implementing the formulations proposed by Bastankhah et al. [35]; secondly, it substitutes the near wake model incorporating the super-Gaussian velocity deficit described by Blondel and Cathelain [36]. These changes result in a faster wake expansion in the near wake region that gradually slows down downstream; furthermore, the wake velocity deficit incorporates itself the combination

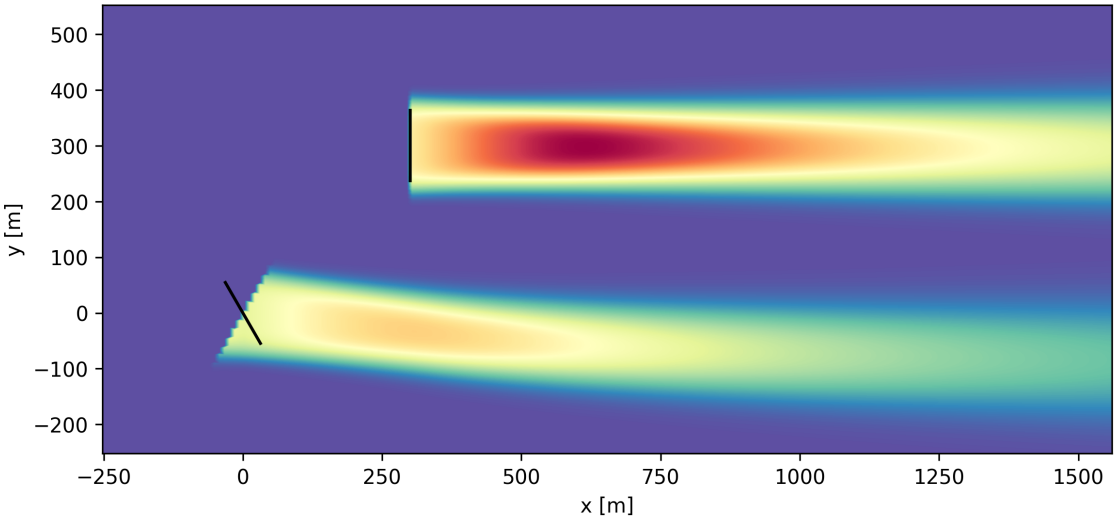


Figure 2.7: Top view of the velocity field in the CC model for both an aligned and yaw-misaligned wind turbine when the wind blows from the west. The plot, obtained in the FLORIS framework, is only intended to give a schematic view of the wake shape.

model, eliminating the need for a separate superimposition model.

The velocity deficit is defines as:

$$U - u = UC_n e^{-\frac{1}{2}\frac{\hat{r}^m}{\sigma_n^2}} \tag{2.29}$$

Here $\hat{r}$ is the radial distance from the wake center normalized with the turbine diameter, σ_n is the defined as

$$\sigma_n = k\frac{x - x_j}{D} + \varepsilon \quad \text{where} \quad k = a_s I + b_s \tag{2.30}$$

where a_s and b_s are model parameters and ε is function of the thrust coefficient as reported in the work by Bay et al. [34]; m is the super-Gaussian order and is a function of the downstream distance from the rotor so that m assumes larger values close to the rotor and smaller values downstream as defined by the relationship

$$m = a_f e^{b_f \frac{x - x_j}{D}} + c_f \tag{2.31}$$

where a_f, b_f and c_f are model parameters; finally the C_n is the velocity deficit amplitude which includes itself a combination method (for further details see [34]).

2.1.4 Wake combinations models

Different wake combination models are implemented in the FLORIS framework to compute the velocity deficit when multiple wakes superimpose:

- **Freestream Linear Superposition (FLS)**. This model simply add the wake velocity deficits

$$\Delta u = \sum_{j=1}^{N_t} \Delta u_j \tag{2.32}$$

where N_t is the number of turbines whose wake needs to be combined and Δu_j is the original wake velocity deficit of turbine j.

- **Max**. This model define the velocity deficit by selecting the maximum

$$\Delta u = \max_{1 \leq j \leq N_t} \Delta u_j \tag{2.33}$$

- **Sum of Squares Freestream Superposition (SOSFS)** [37]. As its name suggests, the velocity deficit in this combination model is computed using the sum of squares of the velocity deficits:

$$\Delta u = \sum_{j=1}^{N_t} \Delta u_j^2 \tag{2.34}$$

The Jensen and GCH models are generally coupled with the SOSFS model, although the other combination models mentioned above can also be used.

2.2 FLORIDyn

In this section an overview of the FLORIDyn model is offered.

FLOw Redirection and Induction Dynamics (FLORIDyn) is a parametric low-fidelity wake model for dynamic wind farm simulations. The model is designed for wake effects prediction in wind plants, controller performance assessment and can be used for control-oriented studies.

FLORIDyn is derived from the steady-state wake model FLORIS. Both models predicts the mean wind park flow using highly simplified physics at low computational cost but if with FLORIS the unsteady features of the wake are neglected, with FLORIDyn the time delays between changes in the control settings and their effects on the downstream wind turbines are taken into account; these delays are due to the finite time requested for the wake advection in the wind farm. So this dynamic model accounts for the wake unsteady attributes also allowing the simulation of heterogeneous and time-varying wind conditions.

The FLORIDyn model was initially presented in the work of Gebraad and Wingerden [38] as an extension of the steady-state counterpart [25]. For the sake of clarity in the following they will be referred to as the Zone FLORIDyn and the Zone FLORIS models. A new version of the dynamic model was build upon the work by Bastankhah and Porté-Agel [23], known as the Gaussian FLORIS model; this revised model is referred to as the Gaussian FLORIDyn model [11]. The MATLAB code [39] used in this thesis is built upon the latter version.

2.2.1 The Zone FLORIDyn model

Building upon its FLORIS counterpart, the Zone FLORIDyn model combines static nonlinear mappings that delineate the wake velocity profile, utilizing an augmented

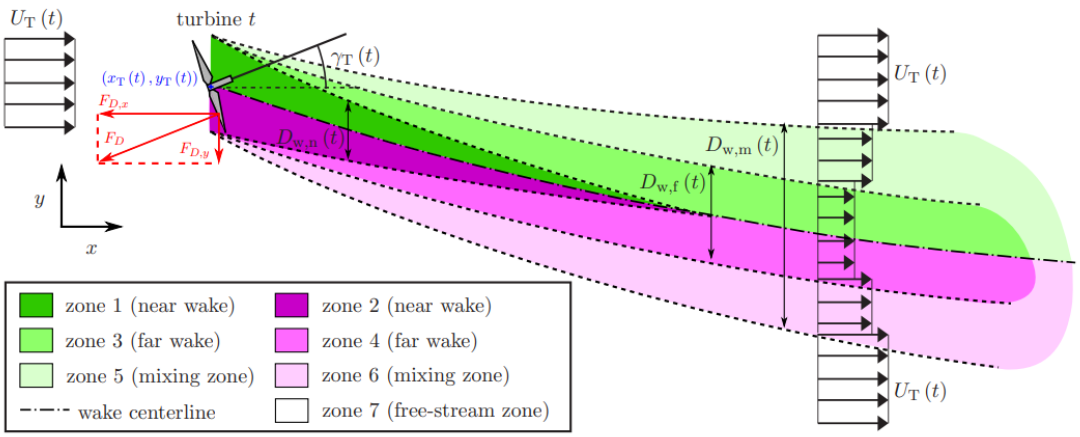


Figure 2.8: Illustration of the zones defined in the Zone FLORIDyn model [38].

Jensen model [27, 40], and the yaw-induced wake deflection, derived from [28]. Furthermore, it incorporates a state-space model that elucidates the advection of control settings changes along the wake.

2.2.1.1 Zones

Adhering to the FLORIS model, in the dynamic model the wake is partitioned into distinct zones, as illustrated in Figure 2.8. Unlike the original steady-state model, FLORIDyn further subdivides the three zones (near wake, far wake and mixing zone) into left and right segments. Additionally, a new zone is introduced to characterize the free stream, resulting in a total of 7 wake zones.

2.2.1.2 Observation points

Within each wake zone of every wind turbine, a number of Observation Points (OPs) is defined; these OPs serve as locations for computing local wake characteristics. Thus for each turbine a finite set of OPs are present in the flow field. In particular, the OPs are generated at the rotor plane and propagate downstream with the effective wind speed they represent.

At each time step, a new set of OPs are generated at the turbine location and a mass of air moves from one OP to the next one downstream. Thus, turbine and flow variables measured at the rotor plane are transmitted along the OPs chains using the time update laws written in the following. Subsequently, these variables are utilized to calculate the wake characteristics. Moreover, in this unsteady model the axial distances between the OPs are dynamically adjusted based on the estimated wind velocity between them.

In Figure 2.9 is illustrated the state update mechanism used in FLORIDyn: during each time step, certain variables, such as the turbine yaw angle γ_T, the axial induction factor a_T, and the free stream velocity U_T, either directly measured at the turbines or estimated from turbine measurements, are sequentially transmitted downstream with the convection of the OPs. Utilizing these quantities, local wake properties, including lateral position and velocity, are computed at each OP using

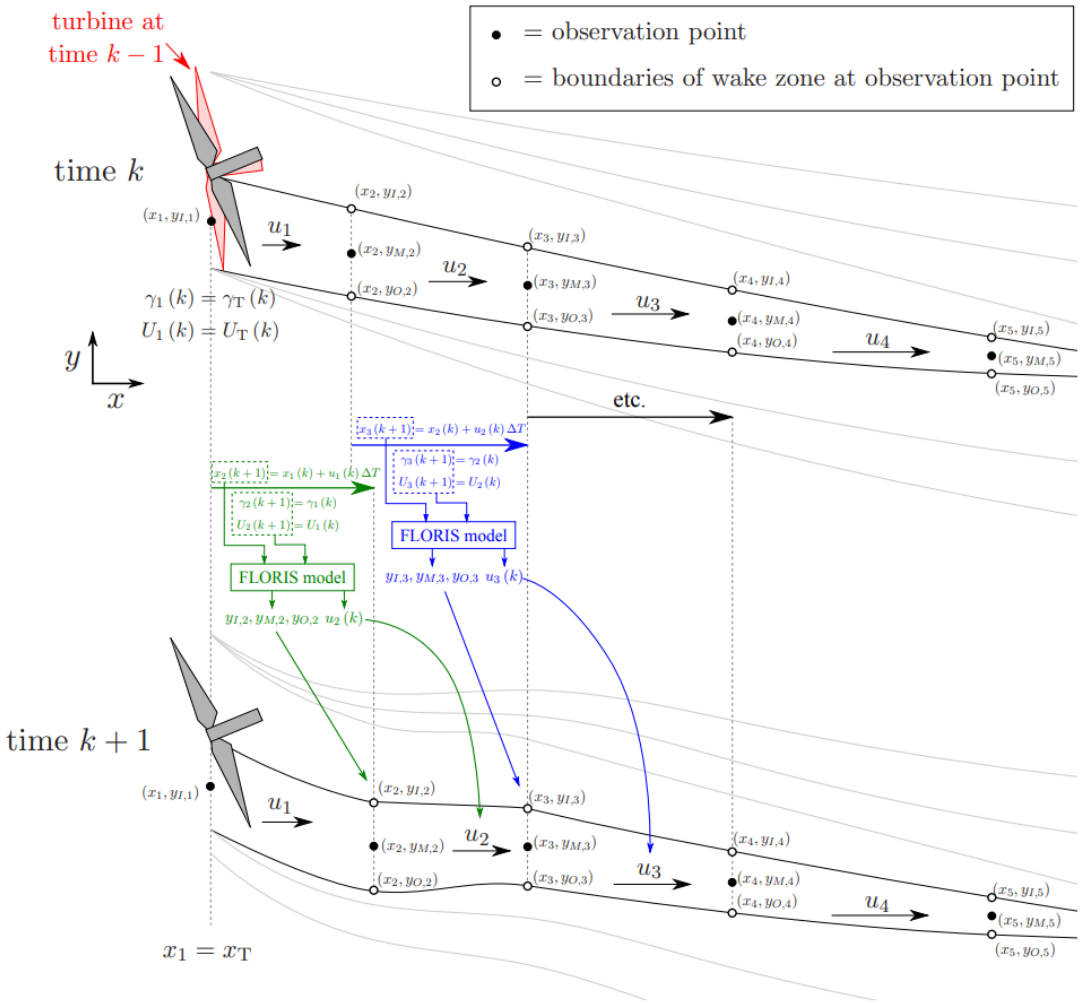


Figure 2.9: Illustration of the state update methodology used in FLORIDyn [38].

the FLORIS model laws. Consequently, at the initial OP, the impact of a yaw change becomes evident after one time step, and at the next OP, the effects of the yaw properties from two time steps ago are observed, and so forth. The same temporal relationship applies to the axial induction and the free stream velocity, creating a cascading observation of their effects along the wake.

2.2.1.3 State update laws

The world coordinates system (x, y) is the one illustrated in Figure 2.8 with x pointing in downwind direction and y along the cross-wind direction. The turbine locations in the wind plant are $(x_T(t), y_T(t))$.

We use the index $z \in \{1, ..., 7\}$ to count the different zones in the wake, $t \in \{1, ..., N_T\}$ to number the turbines in the wind farm with N_T the total number of turbines in the plant. The index k is used to denote the time step. Indicating the total number of OPs in a wake zone with N_P, we use $p \in \{1, ..., N_P\}$ to index the OPs of a turbine in a zone. Thus with $x_{t,z,p,k}$ and $u_{t,z,p,k}$ we indicate respectively the position and the velocity of an OP p in the wake zone z of a turbine t at the

time step k.

Assuming that the velocity between two consecutive OPs is constant along the wake and indicating with ΔK the time step length, the time updated law for the OPs downwind positions is:

$$\begin{bmatrix} x_{t,z,1,k+1} \\ x_{t,z,2,k+1} \\ \vdots \\ x_{t,z,N_P,k+1} \end{bmatrix} = \underbrace{\begin{bmatrix} 0 & & & 0 \\ 1 & 0 & & \\ & \ddots & \ddots & \\ 0 & & 1 & 0 \end{bmatrix}}_{[\boldsymbol{A}]} \begin{bmatrix} x_{t,z,1,k} \\ x_{t,z,2,k} \\ \vdots \\ x_{t,z,N_P,k} \end{bmatrix} + \underbrace{\begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}}_{[\boldsymbol{B}]} x_T(t) + [\boldsymbol{A}] \begin{bmatrix} u_{t,z,1,k} \\ u_{t,z,2,k} \\ \vdots \\ u_{t,z,N_P,k} \end{bmatrix} \Delta K$$
$$\forall t \in \{1, ..., N_T\}, z \in \{1, ..., 7\} \tag{2.35}$$

Adopting the same notations as above, the update state laws for the delayed turbine measurements are:

$$\begin{bmatrix} U_{t,z,1,k+1} \\ U_{t,z,2,k+1} \\ \vdots \\ U_{t,z,N_P,k+1} \end{bmatrix} = [\boldsymbol{A}] \begin{bmatrix} U_{t,z,1,k} \\ U_{t,z,2,k} \\ \vdots \\ U_{t,z,N_P,k} \end{bmatrix} + [\boldsymbol{B}] U_T(t,k) \quad \forall t \in \{1, ..., N_T\}, z \in \{1, ..., 7\} \tag{2.36}$$

$$\begin{bmatrix} \chi_{t,z,1,k+1} \\ \chi_{t,z,2,k+1} \\ \vdots \\ \chi_{t,z,N_P,k+1} \end{bmatrix} = [\boldsymbol{A}] \begin{bmatrix} \chi_{t,z,1,k} \\ \chi_{t,z,2,k} \\ \vdots \\ \chi_{t,z,N_P,k} \end{bmatrix} + [\boldsymbol{B}] \chi_T(t,k) \quad \forall t \in \{1, ..., N_T\}, z \in \{1, ..., 6\} \tag{2.37}$$

with $\chi = \gamma_T$ or a_T. Note that the yaw angle and the induction factor are inherited by the OPs only in zones 1 to 6 since the free stream zone isn't influenced by these turbines variables.

For more details on the equations used in FLORIDyn for the computation of the wake characteristics and on the Zone FLORIDyn model parameters refer to Gebraad et al. [25]. No further information about this version of the unsteady model is given since the framework used in this thesis is built upon the revised version of FLORIDyn.

2.2.2 The Gaussian FLORIDyn model

The Zone FLORIDyn model has some limitations: it cannot account for shear and veer effects due to the two-dimensional flow, its simulations are confined to single wind direction cases, and it does not incorporate turbulent effects since the Jensen velocity deficit model is insensitive to turbulence intensity parameter changes. Additionally, because of the speeds used for the propagation of OPs, there is a possibility of overlapping in certain regions of the wake, leading to a potentially inaccurate computation of the wake.

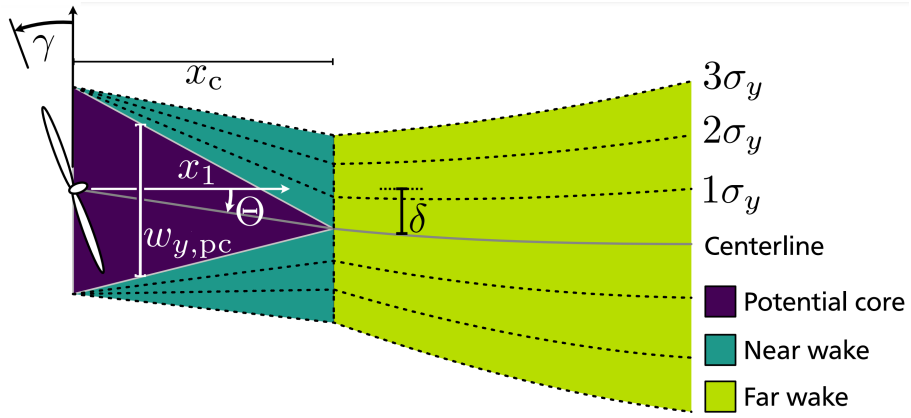


Figure 2.10: Illustration of the 3 regions in the Gaussian wake [11].

In order to overcome this shortcomings, a new version of FLORIDyn was developed by Becker et al. [11] building upon the steady model by Bastankhah and Porté-Agel [23]. This work defines a parametric, three-dimensional wake with a Gaussian-shape velocity deficit in the wake. The unsteady wake model is known as the Gaussian FLORIDyn model.

The Gaussian model was briefly introduced in the description of the GCH wake model in the FLORIS framework (see Subsection 2.1.2), although the formulations differ slightly. For clarity, further details regarding the equations involved are provided here, written in the form as implemented in the FLORIDyn framework.

2.2.2.1 Coordinate systems

Following the nomenclature used in [11], a distinction is made between the global coordinate system (x_0, y_0, z_0) and the wake coordinate system (x_1, y_1, z_1). In the former x_0 y_0 and z_0 are respectively the longitudinal, latitudinal and vertical axes in the global reference system. Whereas, in the (local) wake coordinate system x_1 points in the downwind direction, y_1 in the horizontal cross-wind direction and z_1 in the vertical direction; moreover, in the local system the rotor centre is always located at $(0, 0, 0)$.

2.2.2.2 Regions in the Gaussian FLORIDyn

The Gaussian model identifies 3 regions in the wake, as illustrated in Figure 2.10: potential core, near wake and far wake region. It is emphasized that, unlike what is described in Subsection 2.1.2, here the near wake zone does not include the potential core. Each area is characterized by the reduction factor $r = \Delta u/U$ where U is the free stream wind speed and $\Delta u = U - u$ is the velocity deficit in the wake at hub height.

2.2.2.3 Potential core and near wake

By definition, the potential core is a cone-shape region where the velocity is uniform. After the distance x_c the potential core ends and the velocity profile assumes a self-similar Gaussian distribution. According to [23], the reduction factor in the potential

core and in the near wake areas are respectively:

$$r_{pc} = (1 - \sqrt{1 - C_T}) \tag{2.38}$$

$$r_{nw} = (1 - \sqrt{1 - C_T}) e^{-\frac{1}{2}\frac{[(y_1 - \delta) - w_{y,pc}/2]^2}{s_y^2} - \frac{1}{2}\frac{[z_1 - w_{z,pc}/2]^2}{s_z^2}} \tag{2.39}$$

which are valid for $0 \leq x_1 \leq x_c$, where x_c. In the equations above C_T is the thrust coefficient, $w_{y,pc}$ and $w_{z,pc}$ describes the core width in two cross-wind directions, δ is the wake deflection; s_y and s_z defines the shear layer width and varies from 0 at $x = 0$ to $s_y = \sigma_{yc}$, $s_z = \sigma_{zc}$ with

$$\sigma_{yc} = \sqrt{\frac{1}{8}} D \cos\gamma \tag{2.40}$$

$$\sigma_{zc} = \sqrt{\frac{1}{8}} D \tag{2.41}$$

where D is the rotor diameter and γ is the yaw angle.

The length of the potential core normalized with the rotor diameter is

$$\frac{x_c}{D} = \frac{(1 + \sqrt{1 - C_T}) \cos\gamma}{\sqrt{2}[\alpha^* I + \beta^*(1 - \sqrt{1 - C_T})]} \tag{2.42}$$

where α^* and β^* are two of the 13 model parameters and I is the ambient turbulence intensity, defined in Equation 2.9.

2.2.2.4 Far wake

By following [23], the reduction factor in the far wake region is:

$$r_{fw} = \left(1 - \sqrt{1 - \frac{C_T \cos\gamma}{8\sigma_y \sigma_z / D^2}}\right) e^{-\frac{1}{2}\left(\frac{y_1 - \delta}{\sigma_y}\right)^2 - \frac{1}{2}\left(\frac{z_1}{\sigma_z}\right)^2} \tag{2.43}$$

The normalized standard deviations of the Gaussian velocity in the far wake region along y_1 and z_1 are defined as:

$$\frac{\sigma_y}{D} = k_y \frac{x_1 - x_c}{D} + \frac{\cos\gamma}{\sqrt{8}} \tag{2.44}$$

$$\frac{\sigma_z}{D} = k_z \frac{x_1 - x_c}{D} + \frac{1}{\sqrt{8}} \tag{2.45}$$

The parameters k_y and k_z are determined using Equation 2.8.

2.2.2.5 Wake deflection

From [23], the normalized wake deflection equations for near wake, $x_1 \leq x_c$, and for far wake, $x_1 > x_c$, are respectively:

$$\frac{\delta}{D} = \Theta \frac{x_1}{D} \tag{2.46}$$

$$\begin{aligned} \frac{\delta}{D} =& \Theta \frac{x_1}{D} + \frac{\Theta}{14.7}\sqrt{\frac{\cos\gamma}{k_y k_z C_T}}(2.9 + 1.3\sqrt{1-C_T} - C_T) \\ & \times \ln\left[\frac{(1.6+\sqrt{C_T})\left(1.6\sqrt{\frac{8\sigma_y\sigma_z}{D^2\cos\gamma}} - \sqrt{C_T}\right)}{(1.6-\sqrt{C_T})\left(1.6\sqrt{\frac{8\sigma_y\sigma_z}{D^2\cos\gamma}} + \sqrt{C_T}\right)}\right] \end{aligned} \tag{2.47}$$

where the total deflection angle at the rotor, Θ, is calculated as described by Equation 2.12.

2.2.2.6 Atmospheric stability

The wind speed is the only variable that changes in the vertical direction. A power law is used to define the free stream wind speed along z_0:

$$\frac{U(z_0)}{U(z_0 = z_h)} = \left(\frac{z_0}{z_h}\right)^{\alpha_s} \tag{2.48}$$

where $U(z_0 = z_h)$ is the wind speed at hub height, one input for the model, and α_s is the shear coefficient. According to [41], $\alpha_s > 0.2$ could be used when considering atmospheric stable conditions, $\alpha_s < 0.2$ for unstable conditions. However, it is stressed that the power law coefficient does not define the stability condition, which is also influenced by the surface roughness. Examples for typical α_s values, in particular for offshore application, can be found in [42, 43].

As mentioned above the Jensen model, on which the Zone FLORIDyn is based, neglects the effect of the turbulence intensity. Instead the Gaussian model takes into account this influence, also considering added turbulence caused by nearby turbines (Crespo and Hernández [32]).

This is explicated in Equation 2.8, in which the wake expansion parameters are directly related to the ambient turbulence intensity. The turbulence intensity and the added turbulence intensity are described respectively in Equation 2.9 and 2.10

2.2.2.7 Turbine power

For a given turbine in the wind plant, the following formula, derived from [25], is used:

$$P = \frac{1}{2}\rho A U_T^3 C_P \eta (\cos\gamma)^{p_p} \tag{2.49}$$

where ρ is the air density, $A = \pi D^2/4$ is the rotor-swept area, U_T^3 the effective velocity at the rotor plane, C_P is the power coefficient for zero-yaw, p_p is a model parameter for accounting for yaw misalignment effects on power and η is the turbine efficiency.

2.2.2.8 OPs in the revised model

Unlike the Zone FLORIDyn model, the Gaussian one is three-dimensional and has no regions with defined boundaries. Thus in the revised model OPs must be generated with an other methodology; in particular, an algorithm based on the sunflower distribution by Vogel [44] is used.

Now Ops generated at the same coordinate follow each other forming a chain. The number of chains per turbine, n_c, and the number of OPs per chain, n_{OP}, are two parameters of the model.

In the previous version of the FLORIDyn model, OPs moved with their effective wind speed, causing slower propagation of changes in the central wake regions compared to the outer areas where the velocity deficit is less relevant. Consequently, downstream turbines experienced noticeable delays in responding to state changes (as showed from the comparison with SOWFA simulations results [11]), leading to overlapping areas with distinct properties and creating ambiguity in the wake representation. Instead, in the Gaussian FLORIDyn model OPs propagate with the free stream, according to Taylor's frozen turbulence hypothesis [45]. This eliminates the issue of overlapping wake areas, as neighboring OPs now move at the same speed and undergo similar state changes. This modification also reduces computational load since OPs no longer need to calculate the influence of foreign wakes at every time step, except for those at the rotor plane. Indeed the influence of foreign wakes would be used only to determine the OPs effective wind speed, now equal to the free stream velocity. Nonetheless, a trade-off arises: the state changes effects arrive too quickly at downstream turbines, causing sudden adjustments.

Becker et al. [11] also include directional dependency in the wake orientation and so in the OPs propagation. Hence it is introduced a rotational matrix $[\boldsymbol{R_{0,1}}]$ to transform a vector $\boldsymbol{r_1}$, defined in the global coordinate system, into $\boldsymbol{r_0}$, in the local system of a turbine with rotor-centre $\boldsymbol{t_0}$:

$$\boldsymbol{r_0} = \begin{bmatrix} x_0 \\ y_0 \\ z_0 \end{bmatrix} = \boldsymbol{t_0} + [\boldsymbol{R_{0,1}}](\varphi)\boldsymbol{r_1} = \begin{bmatrix} x_{0,T} \\ y_{0,T} \\ z_{0,T} \end{bmatrix} + \begin{bmatrix} \cos\varphi & -\sin\varphi & 0 \\ \sin\varphi & \cos\varphi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ y_1 \\ z_1 \end{bmatrix} \tag{2.50}$$

where φ is the wind direction, assumed uniform in this equation.

In practice, each Observation Point is linked to two position vectors, $\boldsymbol{r_{0,OP}}$ and $\boldsymbol{r_{1,OP}}$, each within its own coordinate system. When OPs are generated the calculation of $\boldsymbol{r_{1,OP}}$ relies on the arrangement of OPs in the rotor plane, followed by the computation of $\boldsymbol{r_{0,OP}}$ using Equation 2.50 with the wind direction at the rotor plane $\varphi_{0,T}$. To update $\boldsymbol{r_{0,OP}}$ from time step k to time step $k+1$, firstly the OP location in the wake coordinate system is updated, from $\boldsymbol{r_{1,OP}}(k)$ to $\boldsymbol{r_{1,OP}}(k+1)$; then the following equation is used:

$$\boldsymbol{r_{0,OP}}(k+1) = \boldsymbol{r_{0,OP}}(k) + [\boldsymbol{R_{01}}](\varphi_{0,OP})[\boldsymbol{r_{1,OP}}(k+1) - \boldsymbol{r_{1,OP}}(k)] \tag{2.51}$$

Since each OP propagates with its own wind direction $\varphi_{0,OP}$, in the Gaussian FLORIDyn model it's also possible to simulate heterogeneous wind speed cases. Figure 2.11

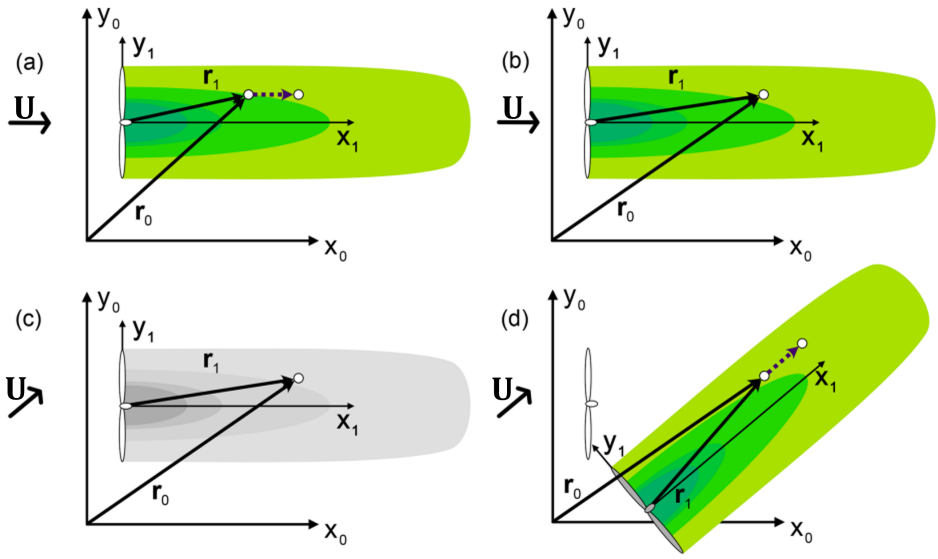


Figure 2.11: Visualization of one OP step in the wake and world coordinate systems in case of non-uniform wind direction [11].

visualizes how Equation 2.51 works; the position vectors index OP is omitted for simplicity. In Figures 2.11.a and 2.11.b the wind direction is constant and the OP is convected downstream as indicated by the dotted arrow. Whereas in Figures 2.11.c and 2.11.d the wind direction is changed and thus the wake depicted in Figure 2.11.c is no more accurate and so the wake coordinate system is rotated in Figure 2.11.d coherently with the wind angle rotation. The change in wind direction affects only the rotational matrix $[\boldsymbol{R_{01}}](\varphi_{0,OP})$, which is calculated differently, whereas $\boldsymbol{r_0}$ and $\boldsymbol{r_1}$ remain the same in their own coordinate system.

2.3 UFLORIS

In this section the UFLORIS model by Foloppe et al. [12] is presented. This is an extension of the FLORIS framework accounting for unsteady features like delayed wake advection between turbines in a wind farm, time-varying and spatially heterogeneous wind conditions.

In the following an overview of the algorithm used in UFLORIS is given, also highlighting the differences and similarities with the Gaussian FLORIDyn model. Then a comparison between the two versions of the dynamic model, UFLORIS corrected and "uncorrecte", is provided.

Except where otherwise indicated, all formulations and theoretical dissertations are derived from [12].

2.3.1 Outline of the simulation algorithm

UFLORIS falls into the low-fidelity wake models class to which also FLORIDyn belongs. It is an analytical parametric model, which requires calibration; exploiting

highly simplified physics it allows to compute the dynamic wake field in wind farms at low cost.

This new dynamic model, developed at the *von Karman Insitute for Fluid Dynamics*, is inspired by the FLORIDyn model [11, 38]. The basic idea is the same: at each time step, Observation Points (OPs) are generated and transported along the wake, thus defining the unsteady field downstream of a wind turbine.

However, there are some differences between the two engineering models. Firstly, whereas in FLORIDyn a number n_c of OPs are generated at the rotor plane for each turbine, in UFLORIS at each time step only one OP per turbine is created at the rotor centre; so instead of utilizing multiple chains of OPs to span the entire wake, the model employs a single chain which defines the wake centerline. In the second place, in UFLORIS the OPs are convected with the velocities computed at their location while in FLORIDyn they travel at the free stream speed. Even if the core of FLORIDyn and UFLORIS is the same, differences can be found in the implementation of the laws. Indeed, UFLORIS exploits directly the FLORIS three modules of the wake velocity, wake turbulence and wake combination only changing the way the unsteady wake centerline is computed.

We use the index $p \in \{1, ..., N_P\}$ to index the OPs of a turbine and $t \in \{1, ..., N_T\}$ to number the turbines in the wind plant; k indicates the time step. Thus, $x_{t,p,k}$ and $\boldsymbol{U_{t,p,k}} = (u_{t,p,k}, v_{t,p,k})$ denote respectively the location and the velocity vector in the horizontal plane of an OP p of a turbine t at the time step k. Indicating with ΔK the time step interval and with $(x_C(t), y_C(t), z_C(t))$ the rotor centre, the state update law for the OPs are:

$$\begin{aligned}\begin{bmatrix} x_{t,1,k+1} & y_{t,1,k+1} \\ x_{t,2,k+1} & y_{t,2,k+1} \\ \vdots & \vdots \\ x_{t,N_P,k+1} & y_{t,N_P,k+1} \end{bmatrix} = \underbrace{\begin{bmatrix} 0 & & & 0 \\ 1 & 0 & & \\ & \ddots & \ddots & \\ 0 & & 1 & 0 \end{bmatrix}}_{[\boldsymbol{A}]} \begin{bmatrix} x_{t,1,k} & y_{t,z,1,k} \\ x_{t,2,k} & y_{t,2,k} \\ \vdots & \vdots \\ x_{t,N_P,k} & y_{t,N_P,k} \end{bmatrix} + \\ + \underbrace{\begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}}_{[\boldsymbol{B}]} \begin{bmatrix} x_C(t) & y_C(t) \end{bmatrix} + [\boldsymbol{A}] \begin{bmatrix} u_{t,1,k} & v_{t,1,k} \\ u_{t,2,k} & v_{t,1,k} \\ \vdots & \vdots \\ u_{t,N_P,k} & v_{t,N_P,k} \end{bmatrix} \Delta K\end{aligned} \tag{2.52}$$

As done in [12], it is assumed that in the cases simulated in this thesis the displacement in the vertical direction of the OPs is negligible compared to the displacements in the horizontal plane. Therefore, Equation 2.52 reflects a two-dimensional dynamic model at the hub height.

The local velocities $\boldsymbol{U_{t,p,k}}$ for all OPs of each turbine are determined through a FLORIS call (steady-state evaluation), considering the background free stream velocity and the turbine settings at time step k.

Then, employing a similar approach utilized for yaw misalignment wake deflection, the wake is adjusted to align with the unsteady wake centerline, which is

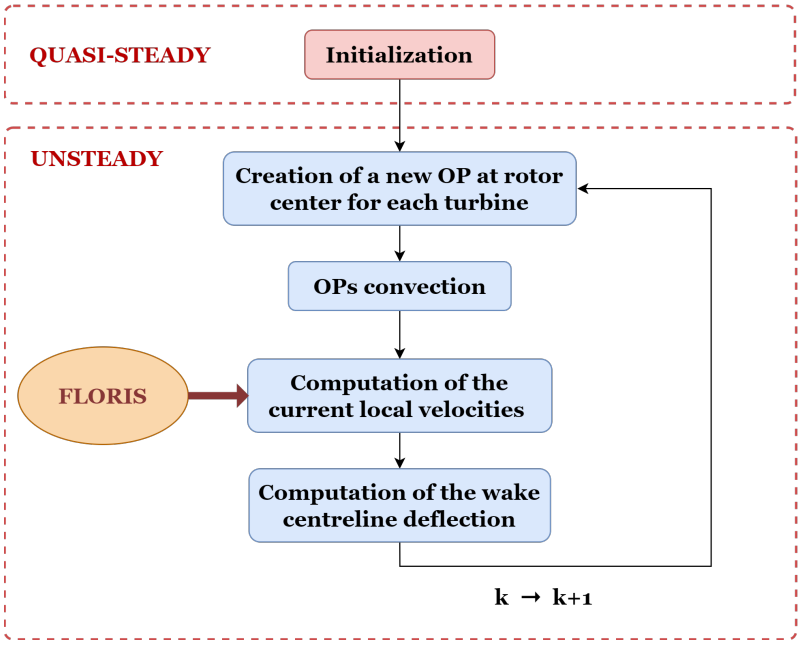


Figure 2.12: Flow chart of the simulation algorithm in UFLORIS.

defined by the OPs.

Disregarding the yaw misalignment, the unsteady wake centerline function is defines as:

$$\delta(x_{t,p,k} - x_{t,1,k}) = (y_{t,p,k} - y_{t,1,k}) \tag{2.53}$$

The steps followed in the UFLORIS model are depicted in Figure 2.3.1. These are:

1. initialization of the simulation;
2. creation of a new OP at the rotor centre for each turbine;
3. convection of all OPs for each turbine using Equation 2.52;
4. calculation of the local velocities with a steady computation (FLORIS);
5. calculation of the wake deflection using Equation 2.53;
6. step forward $k \rightarrow k + 1$, restarting the loop from step 2.

Before updating time, it's also possible to compute the full flow field if a visualization of the horizontal field is requested.

Note that since the total number of OPs, N_p, increases during the initial phase, also the computational cost rises before reaching a plateau.

2.3.2 UFLORIS corrected

Two versions of the model are available: the first one, indicated in [12] as UFLORIS, and the improved one, UFLORIS corrected. Contrary to what is done by Foloppe et al. [12], in this work these model versions are respectively indicated as UFLORIS

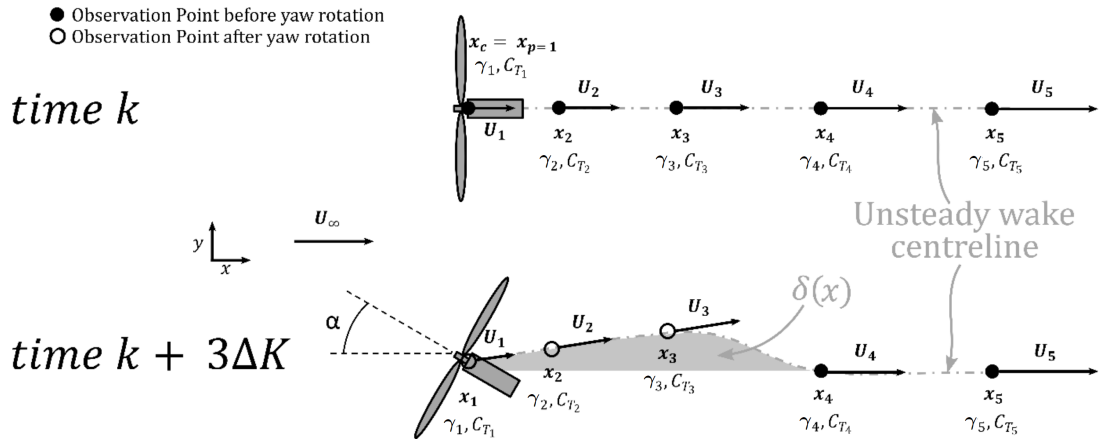


Figure 2.13: Illustration of the state update methodology used in UFLORIS. Since the corrected version of the model is considered, the turbine settings are transported by the OPs and interpolated along the wake to compute the local velocities. A visualization of the unsteady wake centerline deflection is offered. The indices t are omitted in the notation. Figure modified from [12].

"uncorrected" and UFLORIS "corrected" or simply UFLORIS. The improved version of the model accounts for the variation of the turbine yaw angle, γ, and the thrust coefficient, C_T, along the wake. Therefore, two versions have been used in this thesis:

- UFLORIS "uncorrected". During the FLORIS call, the velocity deficit is obtained considering the current turbine settings (γ, C_T). The shortcoming is that this computation isn't compliant with an unsteady approach.

- UFLORIS, the corrected model. In this version the OPs transport with them the turbine settings evaluated at the time the OPs were created. Therefore, the local velocities are computed considering the interpolations of closest γ and C_T.

In Figure 2.13 an illustration of the state update mechanism used in UFLORIS (corrected) is offered. After time k the turbine is yawed of an angle γ; after three time steps only the OPs created after the yaw rotation follow the deflected path. This is how the delayed wake advection is accomplished. Moreover, each OP transports the turbine settings, allowing the computation of the velocity field interpolating the closest γ and C_T.

2.4 FAST.Farm

FAST.Farm is a nonlinear time-domain multi-physics engineering tool designed for predicting the power performance and structural loads of wind turbines within a wind farm [24]. It is developed by the the National Renewable Energy Laboratory (NREL) as an extension of the OpenFAST framework [46]. The latter is a multi-physics, multi-fidelity tool for simulating the coupled dynamic response of wind

turbines, which can be modeled for land or offshore application, on fixed-bottom or floating substructures.

Hence, utilizing the OpenFAST tool, FAST.Farm solves the aero-hydro-servo-elastic dynamics for each turbine in the plant, incorporating additional physics for wind farm-wide ambient wind, a wind-farm super controller, and considerations for wake velocity deficit, advection, deflection, meandering, and merging.

FAST.Farm can be classified as a medium-fidelity wake dynamic modeling frameworks, ensuring accurate predictions while maintaining low computational cost.

In FAST.Farm the wake is modelled based on some of the principles of the Dynamic Wake Meandering (DWM) model [14] but it also deals with some of its limitations. More details are given in Subsection 2.4.4.

FAST.Farm is composed of multiple submodels representing different physics domains of the wind farm. The submodels hierarchy, depicted in Figure 2.14, includes the Wake Dynamics (WD) model, the Ambient Wind and Array Effects (AWAE) model, the super controller (SC) and the OpenFAST (OF) module.

Globally FAST.Farm has four modules and one driver. For each wind turbine in the farm one OF and WD instance is needed, wheres only one SC and AWAE module is connected to the driver for the whole wind plant.

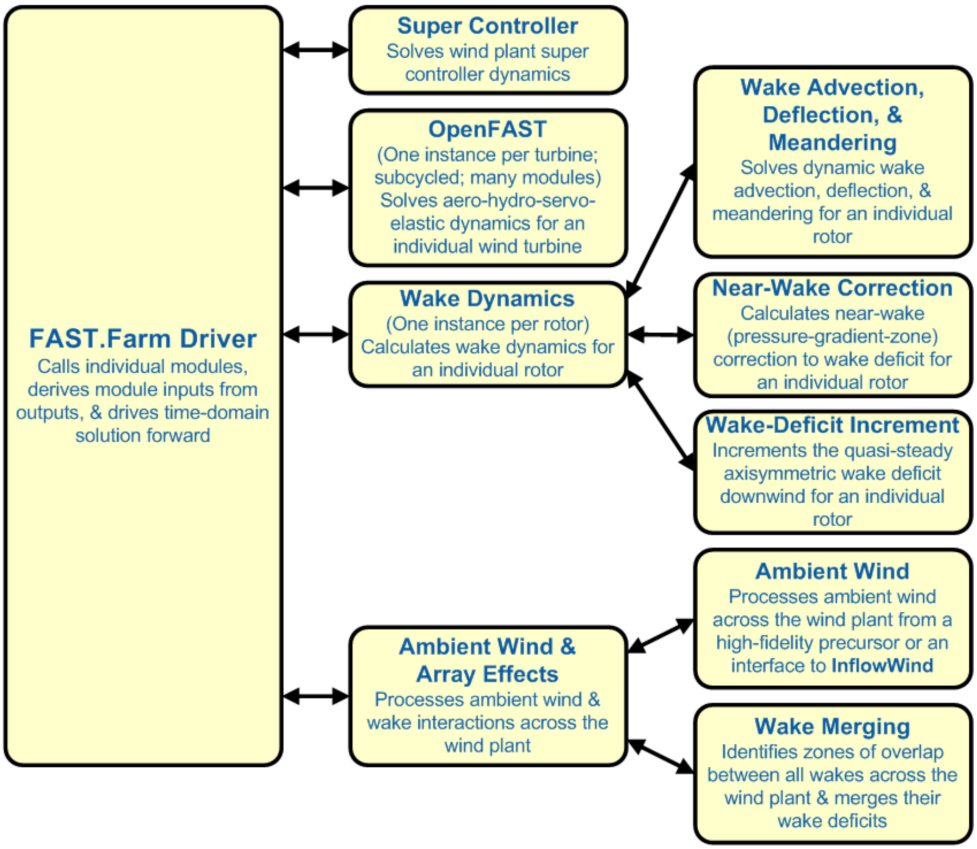


Figure 2.14: FAST.farm submodel hierarchy [24].

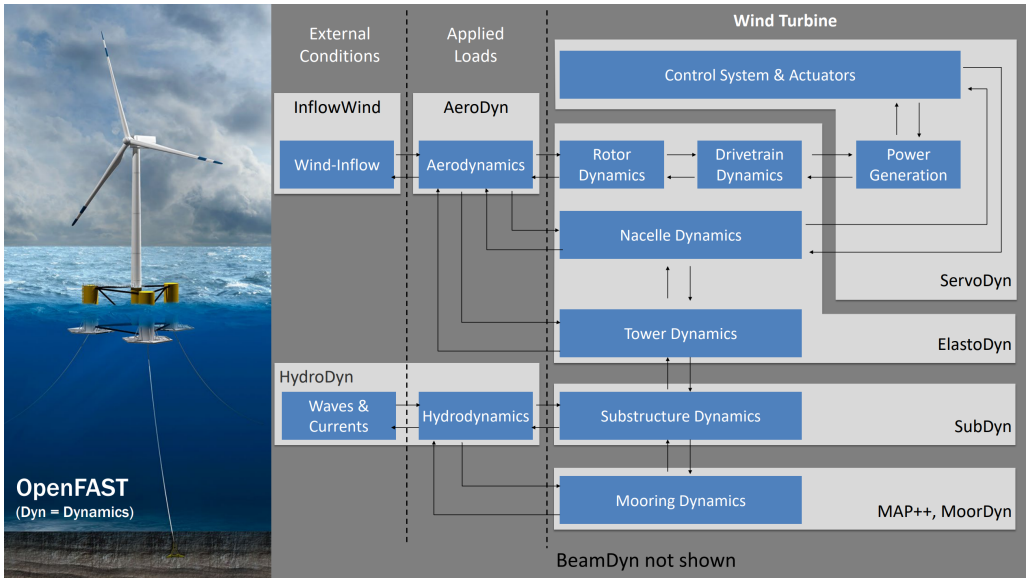


Figure 2.15: Modules within OpenFAST [47].

2.4.1 FAST.Farm Driver

The FAST.Farm driver, often referred to as the "glue code", serves to integrate the multiple modules and drives the time-domain solution forward. Furthermore, this driver reads a simulation parameter input file, validates the parameters, initializes the modules, records results to a file, and deallocates memory upon completion of the simulation.

2.4.2 Super Controller module

Within FAST.Farm, the Super Controller module empowers users to incorporate wind farm-wide control strategies, enabling the exchange of commands between the individual turbine controllers within OpenFAST. So the SC module allows the users to implement directly their own wind farm-wide control strategies which possess the capability to enhance overall power output and mitigate turbine loads on a global scale. This could be achieved, for example, by adjusting wake deficits through modifications in blade pitch or generator torque, and/or by redirecting wakes via variation in nacelle yaw or tilt angles.

Inputs to the SC module consist of commands from individual turbine controllers, sourced from the output of the OF module; conversely, outputs from the SC module comprise global controller commands and commands for individual turbine controllers, serving as inputs for the OF module.

2.4.3 OpenFAST module

The OpenFAST module is a wrapper that guarantee the integration of OpenFAST with FAST.Farm. The OpenFAST framework models the dynamics, including loads and motions, of individual turbines within the wind farm; it also captures environmental factors such as wind inflow and, for offshore systems, waves, current, and

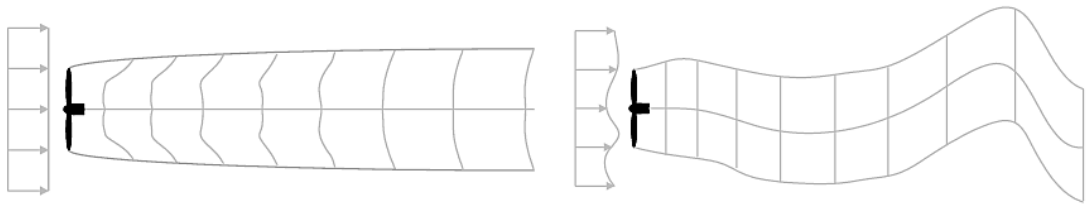


Figure 2.16: Axisymmetric wake deficit (left) and meandered wake (right) [47].

ice, as well as the coupled system response of various components like the rotor, drivetrain, nacelle, tower, controller, and, for offshore systems, the substructure and station-keeping system of each turbine.

OpenFAST is based on different modules corresponding to distinct physical domains within the coupled aero-hydro-servo-elastic solution, as shown in Figure 2.15.

Among all the modules, the noteworthy ones are: AeroDyn, responsible for computing the aerodynamic loads on both the blades and tower; InflowWind which handles the wind inflow within the domain; ElastoDyn which manage the structural properties, initial conditions, and degrees of freedom for the rotor, tower, blades and platform; ServoDyn in charge of regulating the wind turbine based on input from a controller.

2.4.4 Wake Dynamics module

The Wake Dynamics module computes wake dynamics for each turbine, so one instance per rotort is needed in FAST.Farm. It is composed by three modules: the Wake Advection, Deflection and Meandering, the Near-Wake Correction and the Wake-Deficit Increment submodels.

In FAST.Farm the wake is modelled based on the DWM model. This model focuses on capturing crucial wake features, such as the wake deficit evolution for precise wind farm power performance computation and such as wake meandering and wake-added turbulence for turbine loads predictions. This model employs the "splitting of scales" according to which the small turbulent eddies impact wake-deficit evolution, whereas the large ones influence wake meandering. The wake meandering evolution is treated in Larsen et al. [48] by modeling the meandering as a passive tracer that transports the wake deficit transversely, both horizontally and vertically.

In FAST.Farm the passive tracer solution is expanded to include wake deflection and advection and other physical improvements which offer several capabilities: the ability for the wake centerline to be deflected resulting from inflow skew; the capacity for the wake to accelerate from near to far region; changes in wake deficit evolution based on rotor conditions, exploiting a low-pass time filter; the ability for the wake to meander not only transversely but also axially; an elliptical wake shape when looking downwind in skewed flow conditions.

The Near-Wake Correction submodel within the WD module calculates the axial and radial wake-velocity deficits at the rotor disk that are then used as inlet bound-

ary condition for the Wake-Deficit Increment submodel. This correction aims to enhance the accuracy of the far-wake solution by accounting for the decrease in wind speed and radial expansion of the wake in the pressure-gradient zone (or near-wake zone) behind the rotor, factors which wouldn't be addressed using the Wake-Deficit Increment submodel alone.

The Wake-Deficit Increment submodel allows to shift the quasi-steady-state wake deficit downwind. In particular, the wake deficit evolution is based on the thin shear-layer approximation of the RANS equations for quasi-steady conditions in axisymmetric coordinates and with turbulence closure obtained using an eddy-viscosity formulation [49]. In this approximation, the pressure terms are disregarded and the velocity gradients in radial direction are assumed bigger than the ones in the axial direction. Thus the equations of conservation of mass and momentum are respectively:

$$\frac{\partial V_x}{\partial x} + \frac{1}{r}\frac{\partial}{\partial r}(rV_r) = 0 \tag{2.54}$$

$$V_x\frac{\partial V_x}{\partial x} + V_r\frac{\partial V_x}{\partial r} = \frac{1}{r}\frac{\partial}{\partial r}\left(r\nu_T\frac{\partial V_x}{\partial r}\right) = 0 \tag{2.55}$$

where V_x and V_r are the axial and radial velocities and where $\nu_T = \nu_T(x, r)$ is the eddy viscosity. In this thesis no more details are given concerning the wake dynamics formulation, which can be found in the online documentation [24].

2.4.5 Ambient Wind and Array Effects module

The Ambient Wind and Array Effects module in FAST.Farm manages ambient wind and wake interactions within the wind farm, incorporating the Ambient Wind submodel for processing wind conditions and the Wake Merging submodel for identifying zones of wake overlap and for merging wake deficits across the wind plant.

FAST.Farm requires two resolutions of ambient wind: low-resolution for spatial averaging of the wind across wake planes and high-resolution for accurate load computations for individual turbines. The Wake Merging submodel identifies areas of wake overlap in the wind plant and combines wake deficits using the root-sum-squared method for axial components and vector sum for transversal components.

For the calculation, the AWAE module exploits wake volumes, defined as volumes formed by cylinders connecting two consecutive wake planes.

In Figure 2.17 some of the main features of the AWAE modele are illustrated for a simple two wind turbine layout. Wake planes, wake volumes and wake overlap regions are illustrated; the green areas represent the high-resolution ambient wind domains and the yellow dots define the low-resolution domain; the meandered trajectory is defined connecting the center of the wake planes, here indicated as blue points with an arrow representing the orientation of the corresponding wake plane.

The ambient wind data used by FAST.Farm can originate from either a high-fidelity precursor simulation or the InflowWind module in OpenFAST.

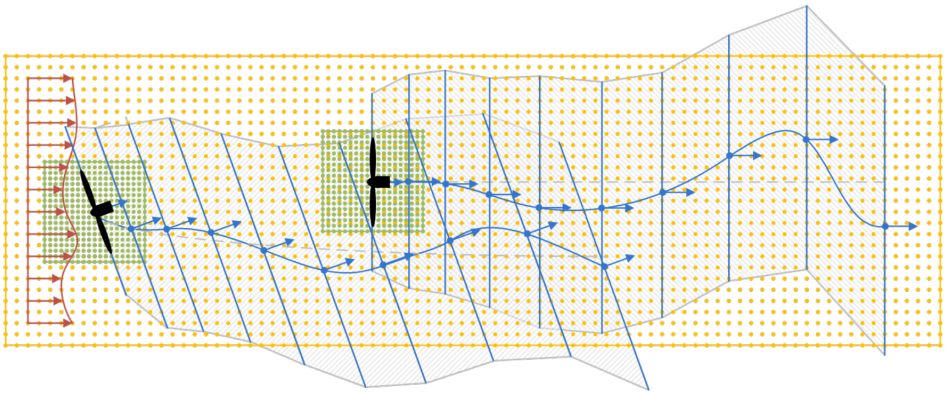


Figure 2.17: Illustration of the meandered wake with identification of wake planes, wake volumes and wake overlap regions [47].

InflowWind allows to generate simple ambient wind scenarios, like uniform wind, or synthetically generated turbulence thanks to, for example, TurbSim. Major details concerning how TurbSim works are given in the following subsection.

Alternatively, ambient wind can be generated by a precursor LES of the whole wind plant, such as the Atmospheric Boundary Layer Solver (ABLSolver) preprocessor of SOWFA [16]. If on one hand a LES ensures a better adherence to physics compared to synthetic turbulence, on the other hand, the use of a high-fidelity precursor requires higher computational cost.

2.4.6 TurbSim

TurbSim [50] is a stochastic, full-field, turbulent-wind simulator tool that is used to generate time series of local wind speeds represented as three-component vectors in a 2D vertical rectangular grid fixed in space. As stated above, its output can serve as input for the InflowWind module in OpenFAST. Specifically, turbulent winds are transmitted through the wind farm based on Taylor's frozen turbulence hypothesis [51]: by interpolating the fields generated by TurbSim, local turbulent wind speeds defined in a 2D plane in TurbSim are used to create a 3D volume in OpenFAST.

TurbSim operates by first defining velocity spectra and spatial coherence in the frequency domain; then, it utilizes an inverse Fourier transform to convert these into time series data.

An important limitation is that it is not possible to prescribe a time-varying wind direction in FAST.Farm using the turbulent inflow generated by TurbSim. The TurbSim parameters $VFlowAng$ and $HFlowAng$ define respectively the mean vertical and horizontal (crosswise) angles of the wind from the X-axis, defined by the parameter $PropagationDir$ in the InlowWind module.

Although the time-series data generated by TurbSim can be post-processed to introduce a time-varying mean wind direction (i.e., defining a time-varying $HFlowAn$), the InflowWind module marches the full field grids along a single propagation direction set in the module, which can only assume a constant value. Consequently,

while velocity vectors can be aligned with the desired variable direction, they are still advanced along a single direction rather than the actual wind direction.

Without modifying the source code, there is no way to obtain accurate simulations with a time-varying wind direction of the TurbSim wind. Instead, it is standard practice to run many OpenFAST simulations (with different inflow parameters) at fixed wind directions.

This is the reason why, in the comparison of FAST.Farm with low-fidelity wake models presented in the next chapter, a test case with a variable wind direction is not considered.

Chapter 3

Test cases

3.1 Comparison of the dynamic wake models

In this section, the intercomparison of the low- and medium-fidelity wake models described in the previous chapter is carried out on a simple three wind turbines in a row layout with time-varying upstream turbine yaw angle control. Firstly, the comparison involves the low-fidelity models FLORIS, UFLORIS and FLORIDyn using a steady wind inflow. Secondly, the engineering models are compared with the medium-fidelity FAST.Farm framework, in which an unsteady turbulent inflow condition is simulated.

The following describes the turbine settings used for the comparison, providing an initial description of the turbine model adopted and instructions for the simulation setup for the low-fidelity models with steady inflow and FAST.Farm in which turbulent inflow is used.

3.1.1 Turbine type

The turbine model used in the simulations of this test case is the NREL 5MW which is a conventional three-bladed upwind turbine developed at the National Renewable Energy Laboratory. This turbine has been employed as a reference in several research projects to establish offshore wind turbine standards and to conduct control-oriented or design-oriented studies [52, 53].

The selection of the NREL 5 MW as the turbine model in the simulations for the intercomparison of wake models is due to the accessibility of technical specifications, including aerodynamic, structural, aeroelastic properties, and control strategies, and to the availability in the turbine libraries of both FLORIS and OpenFAST.

The general specifications of the NREL 5MW are reported in Table 3.1; for further details about aerodynamic and structural properties, the reader is referred to the turbine technical report [54].

In Figure 3.1 it is given the power and thrust coefficient as functions of the wind speed that are used for the power computation with the different wake models. The figure also shows the four regions in which the operation life of a wind turbine is divided. In region I, the wind speed is lower than the cut-in wind speed and is not able to start the rotation of the turbine rotor. Region II lies between cut-in and

Property	Value
Rated power	5 MW
Rotor orientation	Upwind
Blades number	3
Rotor, hub diameter	126 m, 3 m
Hub height	90 m
Cut-in, rated, cut-out wind speed	3 m/s, 11.4 m/s, 25 m/s
Cut-in, rated rotor speed	6.9 rpm, 12.1 rpm
Rated tip speed	80 m/s

Table 3.1: Properties of the NREL 5MW baseline wind turbine [54].

rated wind speed; in this operational zone, the turbine can generate power even though not the nominal one and the control strategies aim at power maximization. In region III, between rated and cut-out wind speed, the pitch control assures the rotational speed to be the nominal one and the wind turbine produces the rated power. Beyond the cut-out wind speed, there is the region IV where the turbine is turned off to avoid structural failures due to too high wind speed.

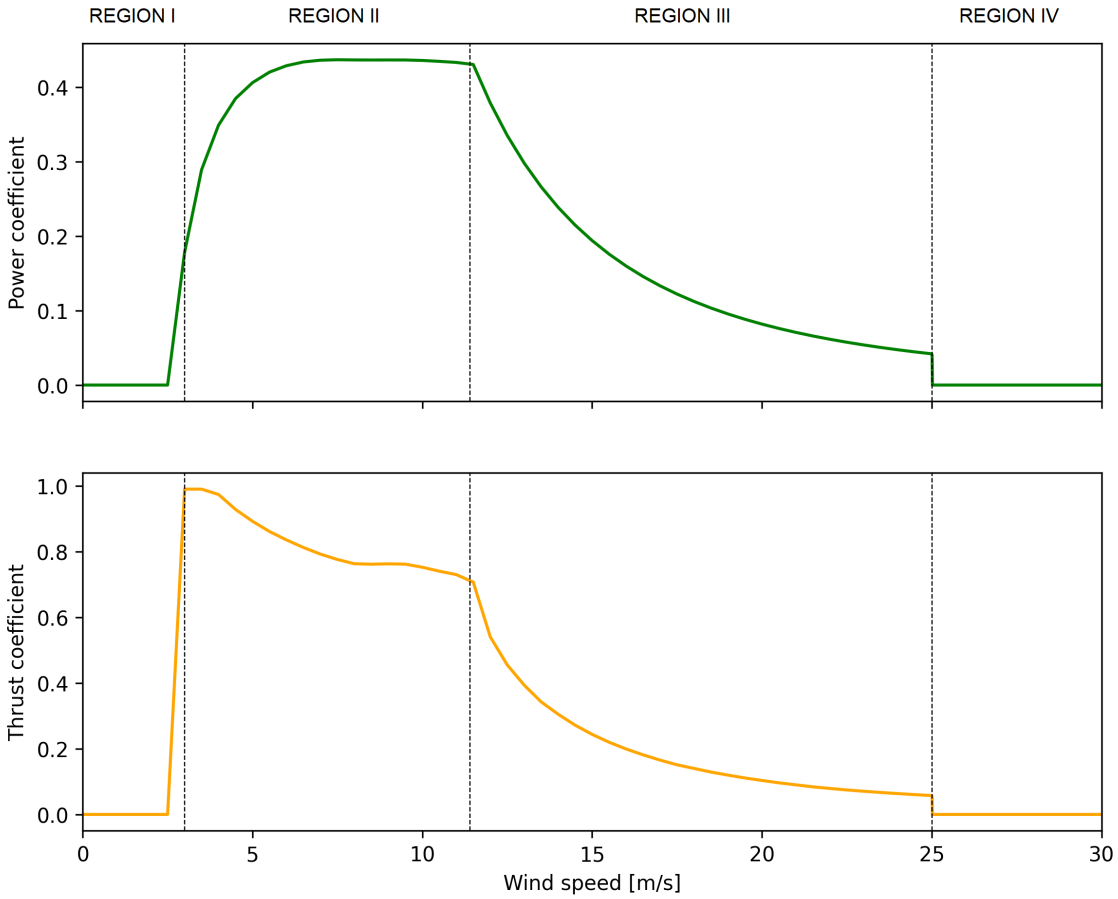


Figure 3.1: Power and thrust curves as a function of wind speed for the NREL 5 MW.

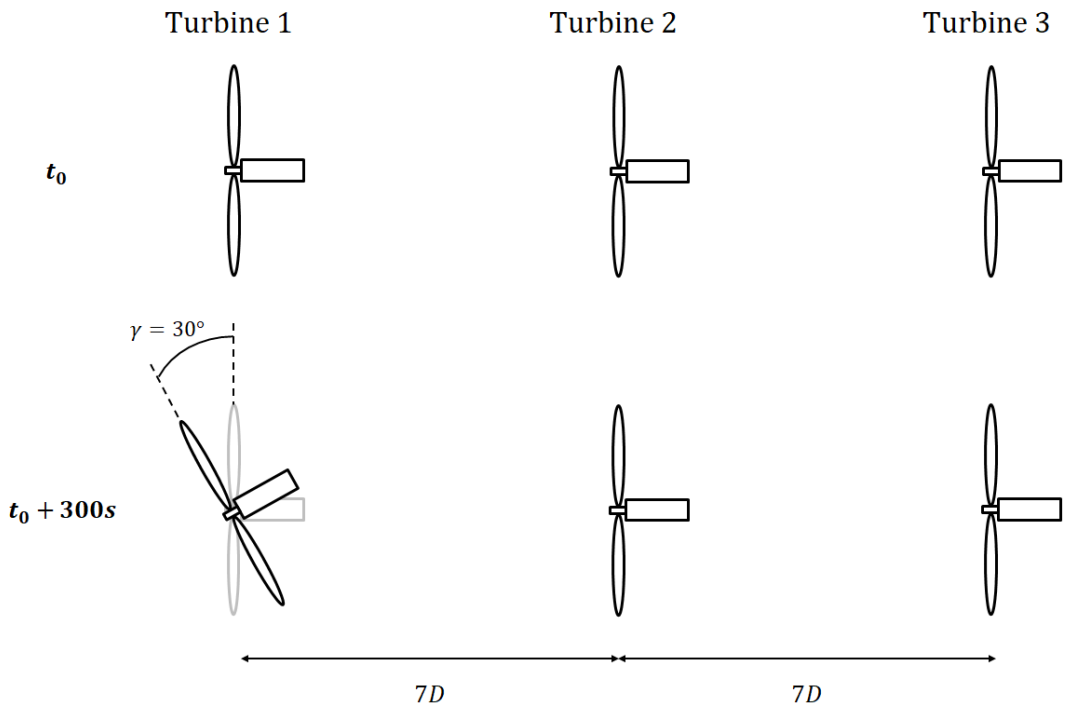


Figure 3.2: Schematic representation of the turbine configuration and yaw control used for the test case; the time t_0 indicates the initialization time.

3.1.2 Low-fidelity models

3.1.2.1 General setup

The simulation is set up using three NREL 5MW turbines in a row, with spacing between them equal to seven diameters. A steady inflow is considered: the wind is blowing from the west at 10 m/s. Regarding the turbine settings, at the beginning of the simulation the three wind turbines are aligned with the wind direction and at 700 s the upstream turbine starts yawing reaching a final yaw angle of 30 degrees in 300 s; after that the upstream yaw angle is maintained for other 600 s. So a total simulation time of 1600 s is considered.

3.1.2.2 Initialization time

A schematic representation of the layout and the yaw setting used for this test case is offered in Figure 3.2, in which t_0 is equal to 700 s and is referred to as the initialization time since no unsteady features are captured.

Indeed, both UFLORIS and FLORIDyn, the two unsteady models, require a certain amount of time during which there are no changes in the background flow and the turbine settings. As described in the previous chapter, in these dynamic wake models, the unsteady features are captured by defining the so-called Observation Points (OPs), which are convected downstream, defining the wake field. The initialization time is selected to ensure that the OPs are transported through the full domain. UFLORIS generally requires a longer initialization time compared to

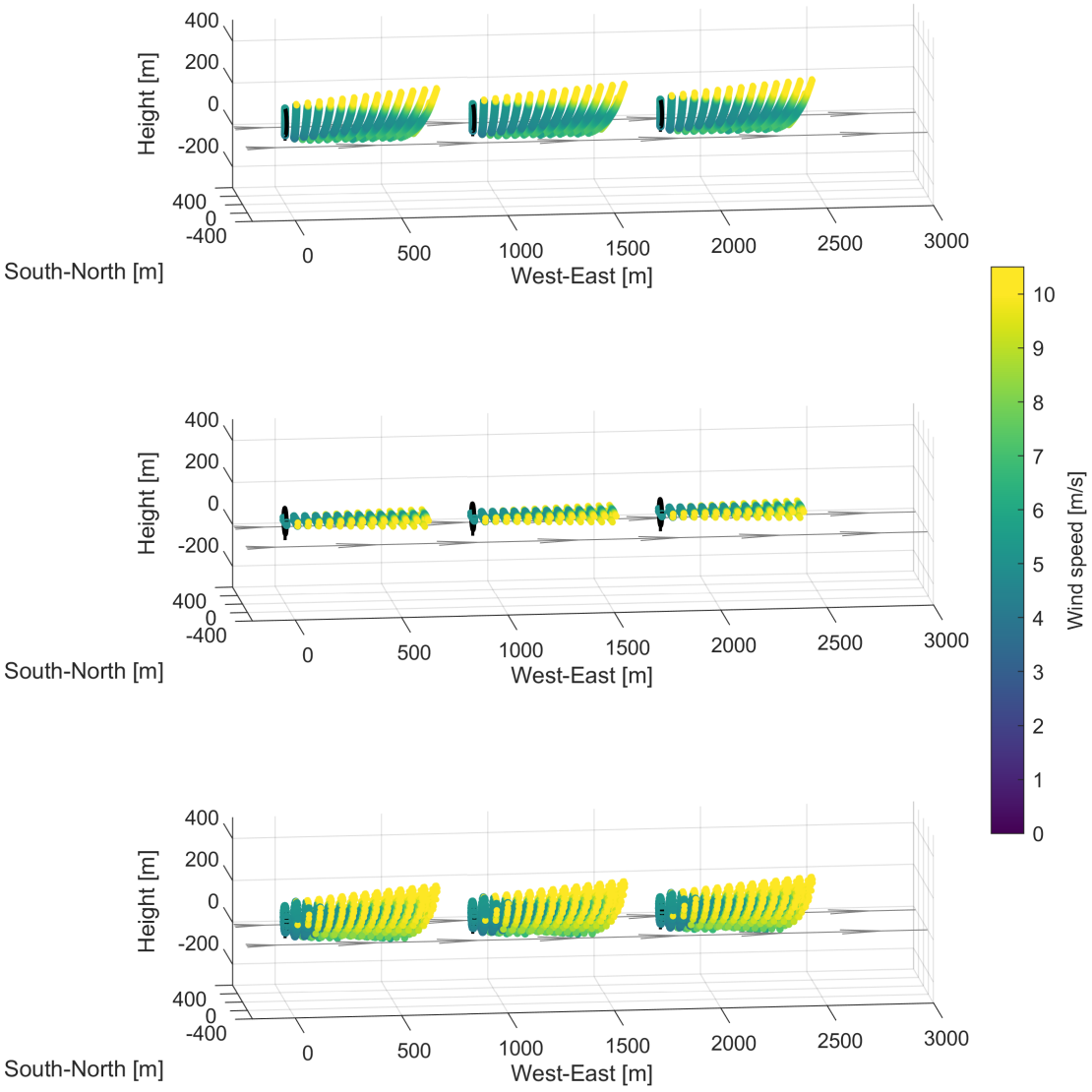


Figure 3.3: Visualization of the available OPs distribution modes within FLORIDyn: from top to bottom there are 2D vertical, 2D horizontal and sunflower distribution modes respectively.

FLORIDyn since in the former, OPs travel at local velocities, which are lower than the wind speed, whereas in the latter, they travel at the free stream velocity. A value of 700 s is sufficient to guarantee a proper quasi-steady initialization of the UFLORIS simulation. Instead, a lower amount of time, such as 300 s, would be needed in FLORIDyn. However, since this test case aims to compare the different solvers, where possible, the same simulation settings are used in the two frameworks, including the initialization time. For the same reason, the quasi-steady computation of 700 s isn't neglected in FLORIS, the steady solver.

Unlike UFLORIS, where a single chain of OPs is generated for each turbine to define the wake centerline, FLORIDyn generates a number of chains n_c of OPs at

the turbine rotor. For the present test case, the default value of $n_c = 100$ [11] is considered. Additionally, since the simulation considers a condition of steady inflow, with the wind always blowing from west to east, a different number of OPs is selected for the three turbines. For the upstream turbine, or turbine 1, each chain contains $n_{OP} = 200$; for turbines 2 and 3, the downstream turbines, $n_{OP} = 150$ and 100 are considered respectively.

3.1.2.3 OPs settings

In FLORIDyn, there are currently three different modes for distributing OPs on the rotor plane: sunrise, 2D horizontal, and 2D vertical. In the 2D vertical distribution mode, OPs are generated and spread across two vertical planes positioned slightly to the right and left of the turbine nacelle. In the 2D horizontal mode, two horizontal planes are defined above and below the hub height. The sunrise distribution mode is based on the Vogel distribution [44] and ensures an equal three-dimensional spread of the OPs at the rotor plane. Figure 3.3 offers a visualization of the various OPs distribution modes during the wake field initialization. It is noteworthy that in the 2D vertical and sunrise distribution modes, OPs above hub height travel faster due to their convection with the free stream speed and the adoption in the simulation of a shear law with a power coefficient $\alpha_s = 0.12$. For a proper comparison with UFLORIS, where a single chain of OPs is generated, and to accurately define the flow field at hub height, the 2D horizontal distribution mode is employed for this test case.

Concerning UFLORIS, a set value for the number of OPs for each turbine is not established. Instead, the default strategy embedded within the framework is employed: OPs generated at time step k are removed once all of them are transported out of the domain. Given the simulation settings, it becomes evident that for this test case, the number of OPs in the individual chain is determined by the OPs from the wake of the upstream turbine.

3.1.2.4 Wake models

For the intercomparison, the Gaussian FLORIDyn framework is used. This unsteady wake model is built upon the Gaussian FLORIS defined in the work of Bastankhah and Porté-Agel [23]. Major details about the velocity deficit, deflection and turbulence submodels are given in Subsection 2.2.2.

To ensure direct correspondence with FLORIDyn, simulations in FLORIS and UFLORIS are conducted utilizing the GCH model [33]. This is an extension of the Gaussian model that incorporates yaw-added wake recovery and secondary steering effect. Further details about the wake model are given in Subsection 2.1.2.

The wake model parameters used in the three simulators are reported in Table 3.2. Different values for the added turbulence parameters are set in the GCH and Gaussian models, as indicated in Table 3.3. All these values were previously adjusted to ensure a fair intercomparison.

Wind field		Wake expansion		Potential core		Power	
I_0	α_s	k_a	k_b	$\alpha = \alpha^*/4$	$\beta = \beta^*/2$	η	p_p
0.06 [9]	0.12 [9]	0.38 [9, 11]	0.004 [9, 11]	0.58 [9, 23]	0.077 [9, 23]	1 [9]	1.8 [9, 38]

Table 3.2: Model parameter values used in three simulators, with references.

Wake model	Added turbulence			
	$k_{f,a}$	$k_{f,b}$	$k_{f,c}$	$k_{f,d}$
GCH	0.5 [9]	0.8 [9]	0.1 [9]	-0.32 [9]
Gaussian	0.73 [11]	0.8325 [11]	0.0325 [11]	-0.32 [11]

Table 3.3: Turbulence parameter values used in the GCH (FLORIS and UFLORIS) and Gaussian (FLORIDyn) wake models, with references.

3.1.3 FAST.Farm setup

Here we describe the test case setup within the FAST.Farm framework, in which turbulent inflow conditions are simulated. In particular, a total of twelve simulations are conducted, considering different synthetically generated turbulence inflows in TurbSim. The purpose of conducting twelve different simulations is to be able to average the results and obtain a statically relevant solution, that can be more readily compared with results obtained from low-fidelity wake models, in which we adopt a steady inflow. This comparison allows for validation of the results obtained with UFLORIS and FLORIDyn.

We begin by providing a broad overview of the model setup; then we describe the selection of the main parameters in the FAST.Farm input file; finally, we outline the settings of TurbSim for generating ambient wind data.

3.1.3.1 General setup

The twelve simulations are conducted using always the same modules input file but involving different wind inflow originated from TurbSim. All simulations use a mean wind speed of 10 m/s at hub height with a turbulence intensity of 6% and a shear power law coefficient equal to 0.12. These values enable a fair comparison with the results obtained from the low-fidelity wake modeling frameworks.

The total simulation time is 1200 s and the upstream turbine yaw control starts at $t = 300$ s. The first 300 seconds of the simulation are omitted since some time is needed for the initialization of the full wake field in the domain and we are only interested in comparing the FAST.Farm results with those obtained with the low-fidelity models starting from the time-varying yaw angle settings for the upstream turbine.

The input files used are described in the following.

- Three OpenFAST input files, one for each wind turbine in the domain. These files guarantee the integration of OpenFAST with FAST.farm, enabling to obtain the aero-hydro-servo-elastic solution for each turbine in the layout.

- One AeroDyn input file. This module is responsible for computing the aerodynamic loads on the blades and on the turbine tower.
- One ElastoDyn input file. This module contains structural properties, initial conditions and flags for the degrees of freedom of the turbine components.
- One InflowWind input file. This module is responsible for defining the wind inflow. For this test case, this input file just calls one of the twelve TurbSim output files which contain the turbulent wind time-series.
- Three ServoDyn input files, one for each turbine. This module defines the modeling options for the controller of the wind turbines. For this test case we set a yaw maneuver of the upstream wind turbine with a rate of 0.1 deg/s from 0 to 30 deg, starting at $t = 300$ s; instead the downstream turbines are always aligned with the west-east direction.
- One FAST.Farm input file. This serves for the definition of the low- and high-resolution domains for the wind farm simulation and the settings of the wake dynamics parameters.
- Twelve TurbSim input files, used to generate the turbulent inflow for the FAST.Farm simulations.

The OpenFAST, AeroDyn, Elastodyn, InflowWind and ServoDyn input files used are the ones written for the NREL 5MW baseline wind turbine that can be found in the GitHub repository of the NREL-supported OpenFAST and FAST.Farm regression tests [55]. The only difference can be found in the yaw control settings within the ServoDyn module, described above. Conversely, tailored parameter values relevant to this test case are configured within the FAST.Farm and TurbSim input files according to the user's guides [24, 50], and are described in the following.

3.1.3.2 FAST.Farm input file

In FAST.Farm, different domains are identified: a large low-resolution domain, in which the wind farm is defined, and several smaller high-resolution domains, one for each wind turbine, in which the aero-servo-elastic response of the single turbine is computed. Following the parameter division as presented in the FAST.Farm input file, we report in Table 3.4 the wind turbines settings, which includes their location and the origin of the high-resolution domains, and in Table 3.5 all the other simulation domain settings.

The positions of the three wind turbines in the low-resolution domain are selected according to what is done with the low-fidelity wake models. The quantities $X0\_High$ and $Y0\_High$, which define the origin of the high-resolution domain, are set equal $1.1D/2$ lower than the turbine location, as suggested in [24]; similarly, $Z0\_High$ is set so that each turbine is contained in its own high-resolution domain.

The low- and high-resolution time steps used for wind data interpolation are selected according to the following inequalities:

$$DT\_Low \leq \frac{C_{Meander} D^{Wake}}{10 V_{Hub}} \approx \frac{C_{Meander} D}{10 V_{Hub}} \tag{3.1}$$

Parameter	Description and unit	Value
WT_X	Location of the wind turbine in x direction [m]	Turbine 1: 0 Turbine 2: 882 Turbine 3: 1764
WT_Y	Location of the wind turbine in y direction [m]	0
WT_Z	Location of the wind turbine in z direction [m]	0
$X0\_High$	Origin of high-resolution spatial nodes in x direction [m]	Turbine 1: -70 Turbine 2: 812 Turbine 3: 1694
$Y0\_High$	Origin of high-resolution spatial nodes in y direction [m]	-70
$Z0\_High$	Origin of high-resolution spatial nodes in z direction [m]	5
dX_High	Spacing of low -resolution spatial nodes in x direction [m]	4.5
dY_High	Spacing of high-resolution spatial nodes in y direction [m]	4.5
dZ_High	Spacing of high-resolution spatial nodes in z direction [m]	4.5

Table 3.4: Wind turbines parameters in the FAST.Farm input file; unless otherwise stated, individual values are relative to all three wind turbines.

Parameter	Description and unit	Value
DT_Low	Time step for low-resolution wind data interpolation [s]	2.2
DT_High	Time step for high-resolution wind data interpolation [s]	0.1
NX_Low	Number of low-resolution spatial nodes in x direction [-]	321
NY_Low	Number of low-resolution spatial nodes in y direction [-]	81
NZ_Low	Number of low-resolution spatial nodes in z direction [-]	31
$X0\_Low$	Origin of low-resolution spatial nodes in x direction [m]	-200
$Y0\_Low$	Origin of low-resolution spatial nodes in y direction [m]	-400
$Z0\_Low$	Origin of low-resolution spatial nodes in z direction [m]	5
DX_Low	Spacing of low-resolution spatial nodes in x direction [m]	10
DY_Low	Spacing of low-resolution spatial nodes in y direction [m]	10
DZ_Low	Spacing of low-resolution spatial nodes in z direction [m]	10
NX_High	Number of high-resolution spatial nodes in x direction [m]	32
NY_High	Number of high-resolution spatial nodes in y direction [m]	32
NZ_High	Number of high-resolution spatial nodes in z direction [m]	37

Table 3.5: Low- and high-resolution domain parameters for the InflowWind module in the FAST.Farm input file.

$$DT\_Low \leq \frac{1}{2f_{max}} \tag{3.2}$$

where $C_{Meandering}$ is a calibrated parameter for wake meandering and is by default 1.9 [24]; D^{Wake}, here approximated as the turbine diameter D, is the wake diameter at the rotor; V_{Hub} is the mean wind speed at hub height (10 m/s); f_{max} is the highest frequency influencing the structural response of the NREL 5MW, which is equal to 2.9361 Hz [54]. It is emphasized that the time step used in OpenFAST is set to 0.01, which is lower than the time steps used in the low- and high-resolution domains; in fact, the former should be smaller than those within FAST.Farm for accuracy and numerical stability reasons.

Indicating with "S" the general axis-direction (X, Y or Z), the spatial discretization parameter values for the low- and high-resolutions domains are selected considering the upper limits:

$$DS\_Low \leq \frac{DT\_LowV_{Hub}^2}{15\,m/s} \tag{3.3}$$

$$DS\_High \leq c_{max} \tag{3.4}$$

where c_{max} is the maximum chord length of the turbine and is equal to 4.652 m [54].

The number of spatial nodes in the three axis-directions for the low-resolution domain are computed as

$$NS\_Low = \text{CEILING}\left(\frac{S_{dist}\_Low}{DS\_low}\right) + 1 \tag{3.5}$$

where the desired domain size are $X_{dist}\_Low = 3200$ m, $Y_{dist}\_Low = 800$ m and $Z_{dist}\_Low = 300$ m. This values are selected simultaneously to the origin of the low-resolution domain.

For the computation of the spatial nodes for the high-resolution domain the following equation is used

$$NS\_High = \text{CEILING}\left(\frac{S_{dist}\_High}{DS\_High}\right) + 1 \tag{3.6}$$

where the lower limit $X_{dist}\_High = Y_{dist}\_High = 1.1D$; for the vertical direction, the following eqaution is used

$$Z_{dist}\_High = HubHeight + \frac{1.1D}{2} \tag{3.7}$$

where the hub height of the NREL 5MW is 90 m [54].

A visualization of the layout with also identification of the low and high-resolution domains is offered in Figure 3.4.

In Table 3.6 we report the wake dynamics parameter values used. These parameters define the axisymmetric finite-difference grid in which each wake plane is discretized and thus influence how the dynamic wake is modelled.

The wake plane radial grid is defined by the parameter dr, which must be selected so that FAST.Farm can accurately resolves the wake deficit within each plane. The following inequality is used

$$dr \leq c_{max} \tag{3.8}$$

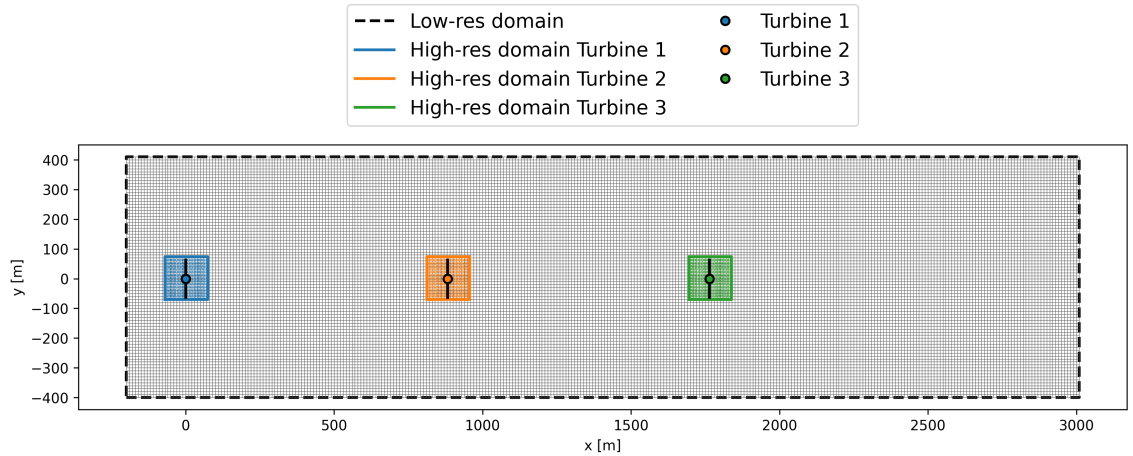


Figure 3.4: Top view of the wind farm layout defined in FAST.Farm, with also the indication of the boundaries of the low- and high-resolution domains.

Parameter	Description and unit	Value
dr	Radial increment of radial finite-difference grid [m]	4.5
$NumRadii$	Number of radii in the radial finite-difference grid [-]	43
$NumPlanes$	Number of wake planes [-]	138
f_c	Cutoff frequency of the low-pass time-filter [Hz]	0.2

Table 3.6: Wake dynamics parameters in the FAST.Farm input file.

The diameter of each wake plane is defined as $2(NumRadii-1)dr$, where the number of radial points is set according to

$$NumRadii \geq \frac{3D}{2dr} + 1 \tag{3.9}$$

The suggested value for the number of wake planes is given by

$$NumPlanes \geq \frac{x_{dist}}{DT\_Low\overline{V}} \tag{3.10}$$

where x_{dist} indicates the downstream distance over which the wake deficit decays, typically set between 10 and 20 diameters, and $\overline{V}$ is the average convection wind speed in the wake, that can be computed as

$$\overline{V} = V_{Hub}\left(1 - \frac{\overline{a}}{2}\right) \tag{3.11}$$

where $\overline{a}$ is the temporal-spatial-mean value of the axial induction factor, here approximated as $1/3$ since we are considering below rated wind speed, as suggested in [24]. Finally, the cutoff of the low-pass time-filter for the wake advection, deflection, and meandering model is set to 0.2, slightly above the value used in example tests in [55].

For the additional parameters that influence the wake dynamics computation, we use the default values [24] which have been calibrated based on SOWFA simulations for the NREL 5MW wind turbine in the work by Doubrawa, Annoni, and Jonkman [56].

Parameter	Description and unit	Value
$NumGrid\_Z$	Vertical grid-point matrix dimension [-]	35
$NumGrid\_Y$	Horizontal grid-point matrix dimension [-]	101
$TimeStep$	Time step [s]	0.1
$AnalysisTime$	Length of analysis time series [s]	1200
$HubHt$	Hub height [m]	175
$GridHeight$	Grid height [m]	340
$GridWidth$	Grid width [m]	1000
$VFlowAng$	Vertical mean flow (uptilt) angle [deg]	0
$HFlowAng$	Horizontal mean flow (skew) angle [deg]	0

Table 3.7: Turbine/model parameters in the TurbSim input file.

3.1.3.3 TurbSim input file

In Table 3.7, the parameter values for the turbine/model specifications section of the TurbSim input file are presented. These parameters dictate the dimensions and shape of the grid where turbulent wind is generated, the frequency content of the time-series data, and the mean wind direction in both lateral and vertical directions.

The selection of the number of grid points in the vertical ($NumGrid\_Z$) and lateral ($NumGrid\_Y$) directions is coordinated with the parameters $HubHt$ and $GridHeight$, which determine the grid dimensions, ensuring a spatial resolution of 10 meters. Note that in this context, $HubHt$ differs from the actual NREL 5MW hub height. This is because TurbSim requires that the $HubHt$ parameter be greater than half the width of $GridHeight$, necessitating the definition of an artificially higher hub height. As specified in the TurbSim user's guide [50], the parameter should satisfy the condition:

$$HubHt > \frac{1}{2} GridHeight \tag{3.12}$$

The values assigned to $GridHeight$ and $GridWidth$ are chosen to ensure that the grid is sufficiently large for the low-resolution domain limits in FAST.Farm.

The parameter $TimeStep$ determines the maximum frequency, $f_{max} = 1/TimeStep$, used in the computation of the inverse Fourier transform to convert velocity spectra and spatial coherence from the frequency domain to the time domain. The parameter $AnalysisTime$ is the length of the data to be analyzed and define the sampling frequency

$$\Delta f = \frac{1}{AnalysisTime} \tag{3.13}$$

Finally, the parameters $VFlowAngle$ and $HFlowAngle$, which define the mean vertical and horizontal flow angle, are set equal to zero as we are considering a mean wind in the west-east direction.

The primary meteorological input parameter values utilized for this test case are presented in Table 3.8. These values are chosen based on the weather conditions simulated in the low-fidelity wake model frameworks. Specifically, we employ a

Input parameter	Description and unit	Value
$IECturbc$	Turbulence intensity in percent [-]	6
$RefHt$	Height of the reference velocity [m]	90
$Uref$	Mean velocity at the reference height [m/s]	10
$PLExp$	Power law exponent [-]	0.12

Table 3.8: Meteorological parameters in the TurbSim input file.

TurbSim simulation	***RandSeed*1**	***RandSeed*2**
I	2073448544	-75384093
II	-1749348676	2090281943
III	-1420799382	836425661
IV	1998735287	605346869
V	-1347477671	-1576098629
VI	1629053665	-1243132589
VII	-1956784321	1067890432
VIII	1258198735	-1906174568
IX	-1432029291	387221162
X	1749011656	-1132166773
XI	1763254892	-982743165
XII	-875421369	1245789632

Table 3.9: Pair of random seeds used for generating the velocity time-series within TurbSim.

power law wind profile with a shear coefficient of 0.12, representing a mean wind speed of 10 m/s at the hub height (90 m), and a turbulence intensity of 6%.

The IECKAI (International Electrotechnical Commission KAImal) spectral model is selected for the simulations. This model is defined in the IEC 61400-1 standards [57, 58], which are the wind turbine design and safety standards published by the International Electrotechnical Commission. Detailed information regarding wind turbine standards and spectral models available in TurbSim is beyond the scope of this thesis, therefore no further details are provided here. Readers are referred to [50] for more comprehensive information on these topics.

The twelve wind time-series data are generated using the same parameter values, as detailed above, in the TurbSim input file. The only two parameters that vary are *RandSeed*1 and *RandSeed*2, which are used together to initialize the pseudorandom number generator (pRNG). The random numbers produced by the pRNG are subsequently utilized to generate random phases (one per frequency per grid point per wind component) for the velocity time series. Keeping the same pair of random seeds allows the user to reproduce the same time-series data. Table 3.9 shows the random seeds used for generating the turbulent wind field in the simulations.

3.2 UFLORIS development

Here, we present a simplified test case for conducting a comprehensive analysis of the unsteady VKI solver, UFLORIS. Specifically, we provide a detailed description of the modifications made to address issues related to the computation of the unsteady wake centerline and wake deficit. Additionally, we propose an alternative strategy for eliminating the OPs that can speed up the simulation while maintaining good accuracy.

The following section is organized as follows: firstly, we provide a description of the baseline test case; secondly, we explain the modifications made in the UFLORIS framework; and finally, we outline the novel OPs deletion strategy.

3.2.1 Experiment

This simple test case involves two NREL 5MW wind turbines spaced seven diameters apart (equivalent to 882 m), with the background flow set to a wind speed of 12 m/s and a turbulence intensity of 6%. Regarding the operational conditions of the turbines, they are initially aligned with the wind propagation direction. At $t = 200$ s, the upstream turbine begins changing between 0° and 30° in cosine-like oscillations with a period of 600 s during 1400 s. Meanwhile, the yaw angle of the downstream turbine remains constant throughout the simulation. Figure 3.5 shows the control settings applied to the upstream wind turbine, highlighting the initialization time, from 0 to 200 s, after which the unsteady computation begins due to the time-varying upstream turbine settings. The total simulation time is 1600 s, and a time step of 10 s is used. The wake deficit is computed using the GCH wake model with parameter values identical to those utilized in the previous test case, as detailed in Tables 3.11 and 3.3.

The choice to consider a case study with a time-varying yaw angle and to use the GCH wake model is related to the issues encountered in UFLORIS under these conditions. Regarding the parameters, other choices could have been made, and these would have led to quantitatively different results. However, the specific values are not of great interest since the present test case aims to provide a simple example for understanding how the modifications made in UFLORIS can enhance the framework by ensuring correct dynamic modeling of wake effects. Moreover, the choice of a simple layout with two turbines is made because the focus is solely on analyzing the unsteady wake of the upstream turbine and how its variations affect the downstream turbine. For the same reason, a similar upstream turbine control as proposed in the work of [12], where UFLORIS is validated against Large Eddy Simulation, is considered here.

3.2.2 Implementation and improvements

Here we propose some corrections to the UFLORIS framework to address the two following issues in the solver.

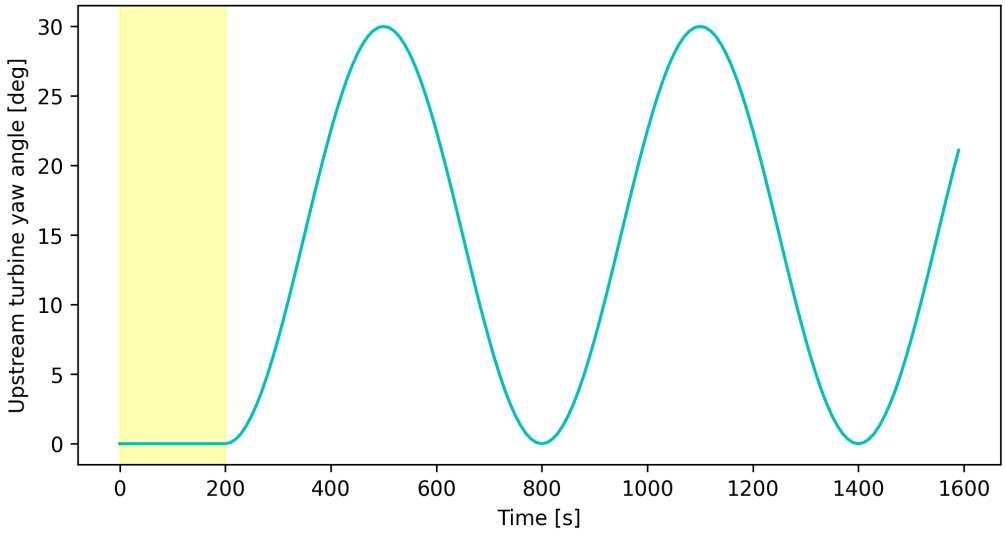


Figure 3.5: Yaw control settings implemented for the upstream wind turbine. The initialization time is indicated in yellow; the unsteady computation starts at $t = 200$ s.

3.2.2.1 Discrepancy with FLORIS in the wake centerline behaviour

The first issue is related to a discrepancy between UFLORIS and FLORIS in the behaviour of the wake centerline when considering steady conditions.

As described in Subsubsection 3.1.2.2, to yield accurate results, UFLORIS requires a certain amount of time, known as the initialization time. During this phase, the OPs are transported through the domain in a steady background field, and the velocity deficit is computed using a quasi-steady approach. One would expect that during this phase, all OPs follow the same steady path coinciding with the wake centerline; particularly, concerning the test case under examination, one would expect straight wake centerline, perfectly aligned with the wind direction. On the contrary, it turns out that during the initialization phase, the OPs deviate significantly from the wake centerline predicted by the steady solver, drifting further as time progresses. Additionally, at the end of the initialization time, when the unsteady approach is activated, the wake centerline is instantly adjusted based on the positions of the OPs, as determined by the algorithm underlying UFLORIS (refer to Section 2.3 for details). This leads to two effects: the lateral deflection of the wakes is overestimated in the initial phase of the unsteady simulation (for this specific case, one would expect zero deflection, as long as the information regarding the yaw angle variation is transported downstream); a jump in the power output of the downstream turbine due to the instant modification of the upstream wake.

Figure 3.6 provides a visualization of the velocity field at the turbine hub height to better understand the issue. In Figure 3.6.a, it is evident that the OPs, indicated by blue dots, follow a different path from the predicted wake centerline during the quasi-steady computation, showing a slight deflection to the right (looking downstream). Conversely, as depicted in Figure 3.6.b, once UFLORIS is activated, the wake centerline of the two turbines is instantly adjusted based on the position of the OPs.

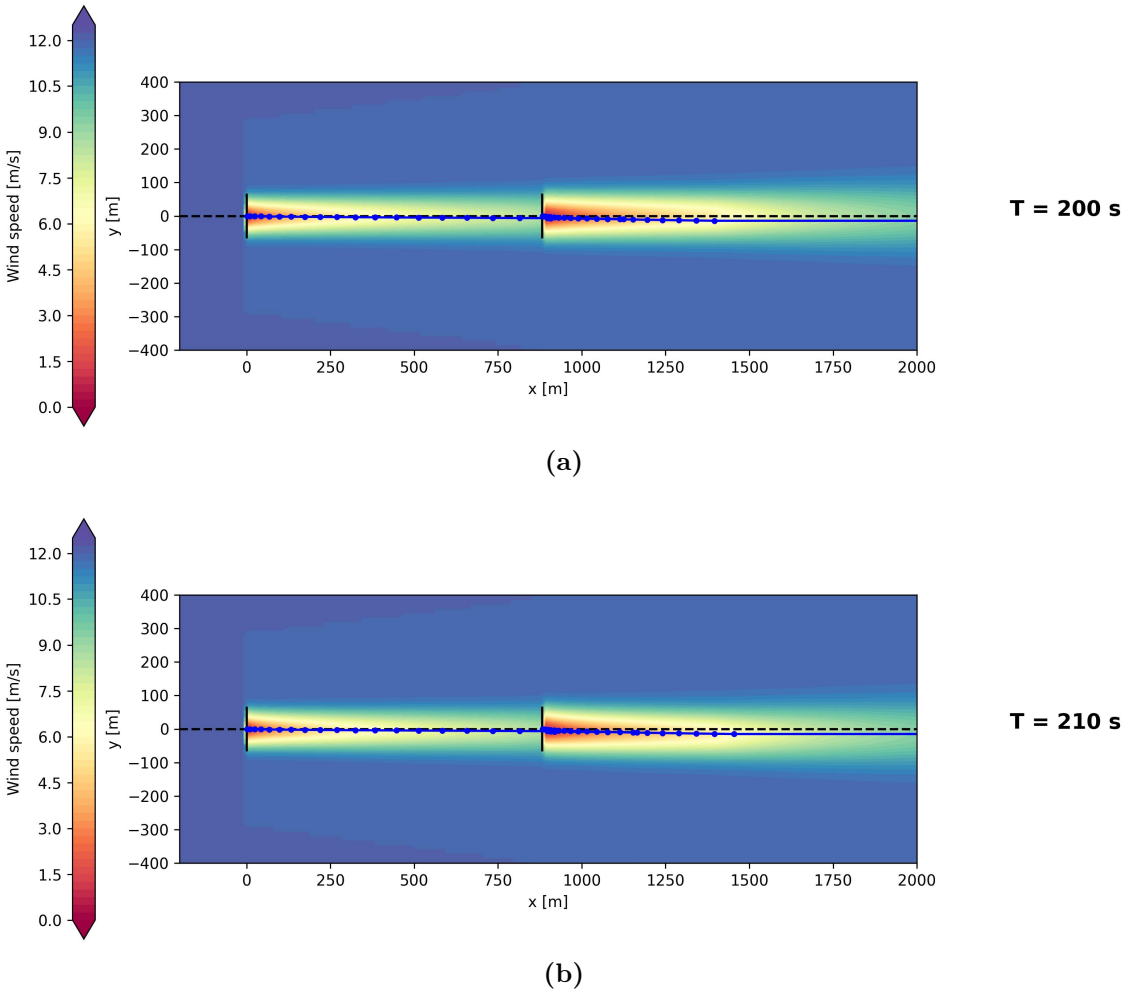


Figure 3.6: Contours of the wind speed at the turbine hub height with visualization of the unsteady wake centerline before (a) and after (b) the initialization time.

This wake centerline behaviour can be attributed to the inclusion of the aerodynamics of the curled wake through transversal wake velocities computation. As detailed in Subsubsection 2.1.2.3, the GCH model (as well as the CC model) calculates spanwise and vertical wake velocities resulting from wake rotation and counter-rotating vortices to account for secondary wake effects. However, because OPs are transported with local axial and spanwise velocities, their positions are also affected by the spanwise wake velocities. In particular, the transversal velocity component responsible for the drifting of the OPs during the initialization time is the wake velocity induced by the ground mirror wake rotation vortex that is defined as

$$V_{wr,g} = \frac{-\Gamma_{wr}(z+z_h)}{2\pi[(y-y_j)^2+(z+z_h)^2]}\left(1-e^{\frac{-(y-y_j)^2-(z+z_h)^2}{\varepsilon^2}}\right) \tag{3.14}$$

where Γ_{wr} is the wake rotation vortex strength (defined in Equation 2.16), z_h is the turbine hub height, y_j is the lateral position of turbine j and ε is usually set equal to two diameters. Indeed, $V_{wr,g}$ is the only non-zero spanwise wake velocity during the initialization phase. As illustrated in Figure 3.7, it takes on negative values,

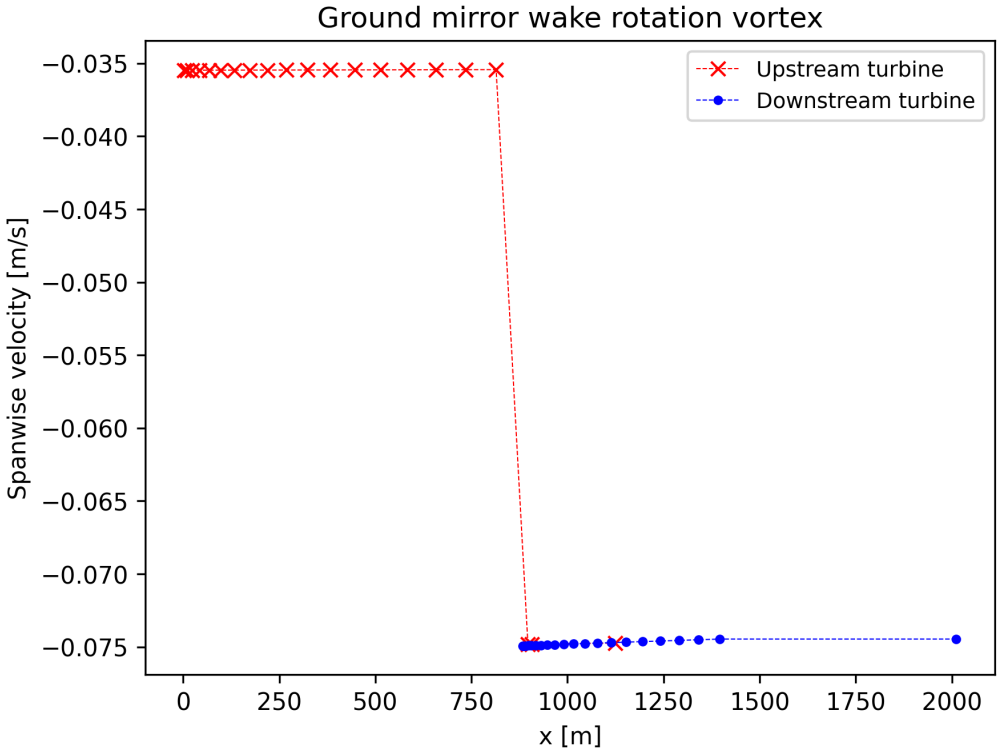


Figure 3.7: Spanwise velocity induced by the ground mirror wake rotation vortex, $V_{wr,g}$, at the OPs axial location computed for the two wind turbines at $t = 200$ s (before the unsteady computation).

leading to the lateral drift of the OPs to the right (when looking downwind).

Therefore, to solve the issue of the discrepancy with FLORIS in the wake centerline behaviour, the computation of $V_{wr,g}$ is disabled, and for consistency, the other two velocity components, in both the two transversal directions, due to ground effect are neglected. Thus, unlike what is done by Martínez-Tossas et al. [31] and King et al. [33], the mirrored vortices below the ground are neglected in the GCH model within the UFLORIS framework and the total spanwise and vertical wake velocities, initially defined as in Equation 2.24 and 2.25, are now computed as

$$V_w = V_{top} + V_{bottom} + V_{wr} \tag{3.15}$$

$$W_w = W_{top} + W_{bottom} + W_{wr} \tag{3.16}$$

For further details about the wake rotation vortex and the counter-rotating top and bottom vortices the reader is referred to Subsubsection 2.1.2.3.

3.2.2.2 Wrong computation of the wake deficit

The second issue involves an improper calculation of the velocity deficit when the unsteady version of the solver is activated, resulting in incorrect dynamic wake modeling. Two clues indicate this problem:

1. at the end of the initialization phase, the delay between the change in the upstream turbine settings and its impact on the downstream turbine is wrongly captured, as the wake shows dynamic effects from the beginning, suggesting the absence of such delay;

2. the power output of the downstream turbine follows a time trend that is inconsistent with the unsteady approach: at the end of the quasi-steady phase, in fact, the power immediately begins to vary and does not show a consistent trend with the turbine settings.

This issue is specifically related to the "corrected" version of UFLORIS (or simply UFLORIS), corresponding to the one used for this test case. As described in Subsection 2.3, UFLORIS in its original version, which is referred to here as the "uncorrected" version, computes the wake deficit considering the turbine settings at each time step. In contrast, in UFLORIS, OPs store the thrust coefficient, C_T, and yaw angle, γ, of their turbines at the time they were created. Subsequently, the velocity deficit at a given position is computed through a steady call using the turbine settings value computed by interpolating the values transported by the closest OPs.

It was found that the error in calculating the velocity deficit is related to improper interpolation of thrust coefficient and yaw angle values. Therefore, to solve this issue, the source code was modified in the following way:

- the `interpolate_ct` function, responsible for interpolating C_T and γ based on the values transported by the OPs (which define the wake centerline), is modified to resolve an implementation error. Algorithm 1 shows the basic corrected implementation of the interpolation function. The first loop serves to determine the accurate index of the turbine in the "flow_field.centerline" array; this is necessary because the turbine index order in FLORIS varies depending on the wind direction, where 0 represents the most upstream turbine. The second loop involves a linear interpolation method based on the downstream distance;

- in the revised UFLORIS source code, the interpolation is performed before the computation of the transversal wake velocities, whereas in the original implementation it was done soon after.

3.2.3 OPs deletion strategy

One of the main causes of the slowdown in UFLORIS is the addition of one dimension, time, to the model. This slowdown becomes particularly significant when considering a great number of turbines in the domain; evidence of this is provided in the latest test case presented in Section 3.3. Even when using a simple test case with two turbines, as in the present case, it becomes evident how the number of OPs influences the computational cost of a simulation in UFLORIS. Thus, a fundamental aspect is to select a proper strategy for the elimination of OPs, based on the specific test case.

Algorithm 2 Interpolation function to determine the thrust coefficient and yaw angle values at a given position in the domain.

procedure INTERPOLATE_CT(x_i, y_i, flow_field, grid)
  Initialize turbine rotors location $x_{\text{rot}}, y_{\text{rot}}$
  for $jj \leftarrow 0$, length(flow_field.centerline) **do**
    $x_{i\_\text{rot}} \leftarrow x_{\text{rot}}[jj]$
    $y_{i\_\text{rot}} \leftarrow y_{\text{rot}}[jj]$
    if $x_i == x_{i\_\text{rot}}$ and $y_i == y_{i\_\text{rot}}$ **then**
      $cc \leftarrow jj$
    end if
  end for
  $x_{\text{centerline}} \leftarrow$ flow_field.centerline[cc][0]
  $y_{\text{centerline}} \leftarrow$ flow_field.centerline[cc][1]
  $x_{\text{locations}} \leftarrow$ grid.x_sorted $- x_i$
  Initialize C_T and γ
  for $ii \leftarrow 0$, length(grid.x_sorted[0, 0, 0, :, 0]) **do**
    for $jj \leftarrow 0$, length(grid.x_sorted[0, 0, 0, 0, :]) **do**
      for $kk \leftarrow 0$, length(grid.x_sorted[0, 0, :, 0, 0]) **do**
        $\gamma[:, 0, kk, ii, jj] \leftarrow \text{interp}(x_{\text{locations}}[:, 0, kk, ii, jj], x_{\text{centerline}},$
        reverse(flow_field.centerline[cc][3]))
        $\gamma[:, 0, kk, ii, jj] \leftarrow \text{interp}(x_{\text{locations}}[:, 0, kk, ii, jj], x_{\text{centerline}},$
        reverse(flow_field.centerline[cc][4]))
      end for
    end for
  end for
  return C_T, γ
end procedure

The default OPs deletion strategy is to remove OPs generated at a given time step when these are outside of the domain for each turbine chain. So, it is clear that this strategy is not the optimal one, specifically when large domains are considered.

Here we present a different strategy: removing the old OPs that have been convected for a period longer than a specified time. Since a new OP per turbine is created at each time step and since the time step is constant (10 seconds in this case), this approach is equivalent to setting a maximum number of OPs for each turbine.

One simulation is done using the default strategy implemented in UFLORIS, which is related to the domain dimensions. For the current test case, the extremes considered are $x = [-200, 2500]$ m in the axial direction and $y = [-400, 400]$ m in the transversal direction, with turbines positioned at $(0, 0)$ m and $(882, 0)$ m. Since the wind direction aligns with the x-axis and the maximum wake deflection upstream is such that it keeps the wake within the transverse boundaries, the only limiting factor for the number of OPs is the upper axial limit of 2500 m. The results of this simulation are used as a baseline for comparison of accuracy.

Then, other seven simulations are run using the novel OPs deletion strategy based on limiting the number of OPs per each turbine. Specifically, the number of OPs used in the different simulations varies from 5 to 35 with an increment of 5.

It is emphasized that, in general, the selection of the number of OPs is not straightforward and must be done taking into account both the layout and parameters such as turbulence intensity and wake expansion. The number of OPs should be chosen to ensure that the unsteady wake centerline is calculated until the wake has largely recovered, and thus its impact on downstream turbines is negligible. In particular, when considering a large wind farm, it is not necessary to ensure the convection of OPs throughout the entire domain. Instead, it is necessary for them to be transported a sufficient distance so that the wake deficit is close to being eliminated. Therefore, the number of OPs should be selected based on the turbulence intensity that influences wake recovery: higher turbulence levels are expected to result in greater wake recovery and hence a smaller number of OPs is required for accurate results. Selecting too few OPs would lead to an incorrect capture of dynamic effects: downstream of the last OP, where the wake has not yet fully recovered, the centerline would be calculated using a steady approach.

3.3 Low pressure event

In this last test case we deploy UFLORIS "at scale". Particularly, we carry out a dynamic investigation of an Extreme Weather Event (EWE) to explore the capabilities and limitations of the wake models within the UFLORIS framework. Due to the large scale of the simulation layout (a total of 572 offshore wind turbines) the goal is to see whether a dynamic wake modeling solver can capture strongly dynamic conditions better than its steady counterpart. Then the simulation results are compared with the SCADA (Supervisory Control and Data Acquisition) data collected at the Belgian offshore wind farms.

The case study is based on the low-pressure system that occurred on December 24, 2020, in the North Sea. The test case is inspired by the work of Vemuri et al. [59]. The selection of this real weather event is motivated by the necessity of a dynamic investigation due to the fast variation in wind speed and direction, which resulted in severe power losses for the Belgian offshore wind farms.

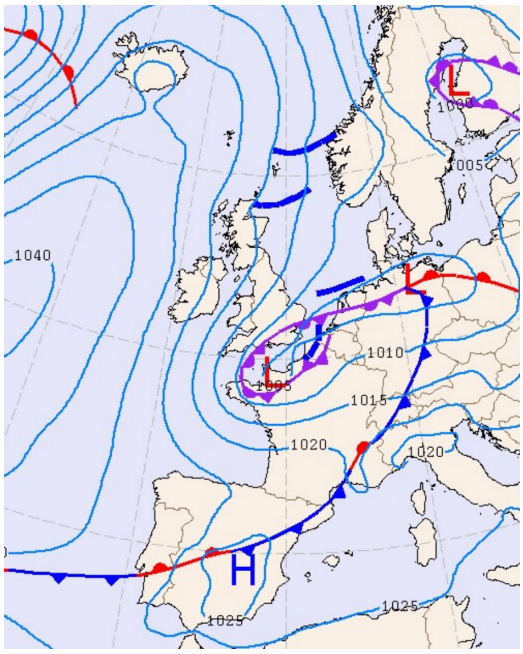
This section is organized as follows: first, a brief description of the weather event is provided; secondly, we describe the farm layout considered in the simulations; next, the wake models used are explained; following that, the simulation setup is detailed; and finally, we explain how the results are post-processed.

3.3.1 Synopsis of the event

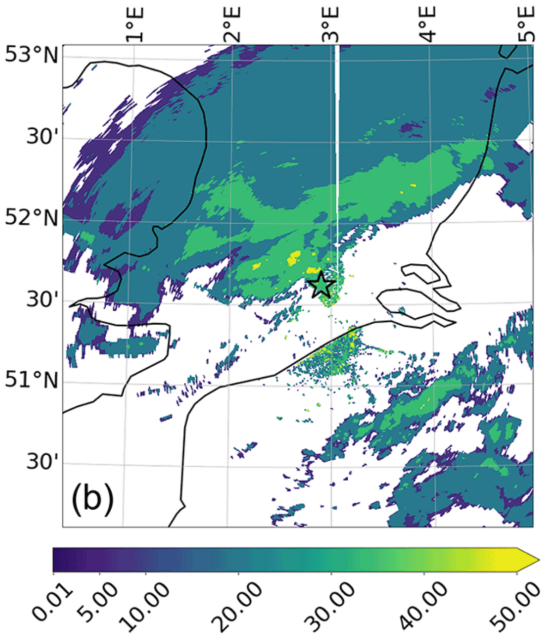
A low-pressure system is a region in the atmosphere where the air pressure is lower than the surrounding areas. Low-pressure systems are usually associated with precipitation as warm, moist air rises within the low-pressure region, cooling and condensing, thus generating clouds and eventually resulting in precipitation. Such events play a crucial role in the operational life of a wind turbine and must be taken into account for estimating loads and power losses.

This test case refers to a real low-pressure system that occurred in the North Sea, thus it is referred hereafter to as *low pressure event*. Specifically, on 24 December 2020, the offshore wind farms in Belgium experienced significant rainfall coupled with rapid shifts in wind speed and direction. The synoptic map in Figure 3.8.a shows the existence of a low pressure system in the North Sea. Radar data from the Royal Meteorological Institute of Belgium (RMI-B) reveal substantial precipitation zones over the Belgian North Sea, as illustrated in Figure 3.8.b. To understand the exceptional nature of the event, consider that the annual rainfall across the North Sea varies between 340 and 500mm, averaging 425mm [60].

SCADA data indicate rapid changes in the wind direction of 100 degrees and in the wind speed of 10 m/s.



(a) Synoptic map at 00:00 UTC (credit of the Royal Netherlands Meteorological Institute).



(b) Observed precipitation rate in mm/h at 02:00 UTC, provided by a C-band Doppler radar located in Jabbeke on the Belgian coast. The star indicates the location of the Belgian offshore platform.

Figure 3.8: Synoptic map (a) and observed precipitation rate (b) for the low pressure event on 24 December 2020 [59].

3.3.2 Farm layout

The Belgian Offshore Platform (BOP) [61] is a non-profit organization representing wind farm owners and investors in the Belgian zone of the North Sea. To date, eight offshore wind farms are installed and operational, divided into nine concession areas. Currently, offshore wind energy in the Belgian North Sea has an installed capacity of 2262 MW and produces approximately 8 TWh of green electricity annually, covering about 10% of the electricity demand in Belgium. By 2030, the offshore wind

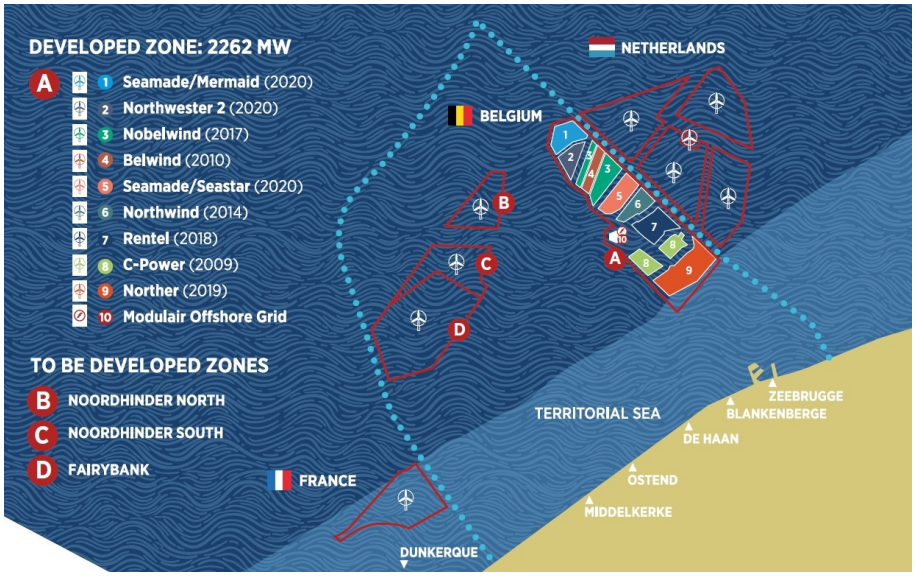


Figure 3.9: Illustration of the Belgian wind energy zone, including both the existing Belgian wind farms (zone A) and the new Princess Elisabeth concession (zones B, C, and D), and of the Netherlands area corresponding to the Borssele wind farm zone [61].

capacity in the North Sea is expected to expand to 6-8 GW, ensuring approximately 30% of the electricity demand [61] .

Although the purpose of the test case is to calculate the power variation of one of the Belgian wind farm due to the EWE, it is necessary to consider not only its turbines but also neighbouring wind farms. Indeed these contribute to additional power losses because downstream turbines operate at lower wind speeds influenced by the wakes of upstream turbines. Therefore the neighbouring farms are considered, including the Dutch farms in the Borssele wind farm zone [62].

Figure 3.9 depicts the entire Belgian-Dutch cluster examined in the case study. In addition to the existing Belgian wind farms (zone A) and the Borssele wind farm zone in the Netherlands, it also shows the Princess Elisabeth zone (B, C, D), which has not yet been developed and so is not included in this case study.

Details about the offshore wind farms included in the test case are reported in Table 3.10.

3.3.3 Models within the UFLORIS framework

For the case study, a total of four simulations are conducted using different models within the UFLORIS framework, to highlight their differences.

In the first simulation, UFLORIS is utilized in its "uncorrected" version, where the velocity deficit at the i-th time step is obtained considering the thrust coefficient values of the same time step. For this simulation, the default strategy for OPs elimination is employed: OPs generated at the i-th time step are removed when all OPs from that time step are transported out of the computational domain.

Two simulations are conducted using UFLORIS in its "corrected" version. In these simulations, OPs carry the thrust coefficient and yaw angle values correspond-

Wind farm	Installed capacity [MW]	Turbines
Norther	370	44 × Vestas V164-8.4MW
C-Power	325	6 × REpower 5MW 48 × REpower 6.15MW
Rentel	309	42 × Siemens SWT-7.0-154 7.35MW
Northwind	216	72 × Vestas V112-3MW
SeaMade - zone Seastar	252	20 × Siemens Gamesa SG 8.0-167 DD 8.4MW
Nobelwind	165	50 × Vestas V112-3.3MW
Belwind	171	55 × Vestas V90-3MW 1 × Alstom Haliade 150 6MW
Northwester 2	219	23 × Vestas V164-9.5MW
SeaMade - zone Mermaid	235	28 × Siemens Gamesa SG 8.0-167 DD 8.4MW
Borssele I&II	752	94 × Siemens Gamesa 8 MW
Borssele III&IV	731.5	77 × Vestas V164 9.5 MW
Borssele V	19	2 × Vestas V164 9.5 MW

Table 3.10: Belgium and Dutch offshore wind farms considered for the simulation. Data from [61, 63–65].

ing to the instant they were created, which are then utilized via interpolation for wake field calculation. Interpolation of C_t and γ constitutes one of the main sources of slowdown in UFLORIS. Given the high number of time steps considered for the case study, a different strategy for OPs elimination is adopted compared to UFLORIS "uncorrected". Specifically, in the two simulations, OPs are eliminated once they have been transported for distances greater than 3 km and 5 km, respectively. This strategy differs slightly from the one described in Subsection 3.2.3 because, given the significant variability in wind speed for this EWE, fixing the length of the OPs chain is more suitable than specifying a fixed number of OPs.

Finally, a simulation is conducted using a quasi-steady approach, hence the term QS FLORIS. In this simulation, OPs are not generated as the dynamic nature is neglected, and at each time step the velocity deficit is obtained through a call to FLORIS.

3.3.4 Simulation setup

The full zone layout includes a total of 572 offshore turbines. The layout used for the simulation is represented with circular markers in Figure 3.10.

In the simulations done within the UFLORIS framework, a time step of 15.06 s is utilized. This value is chosen based on the wind speed and direction data used to define the background flow, ensuring a constant time step that is a fraction of the smallest time interval between two consecutive wind data points. The initialization time, which involves convecting observation points using a quasi-steady approach, is set to 1505.98 s. This duration ensures that every OPs chain is fully established and each turbine wake is properly initialized. The simulation time is defined as 24699 s. Hence, the total simulation time in UFLORIS amounts to 26204.98 s (approximately 7 hours and 17 minutes), spanning across 1740 time steps.

A yaw error of 0 was assumed at each instant for each turbine. The motivation

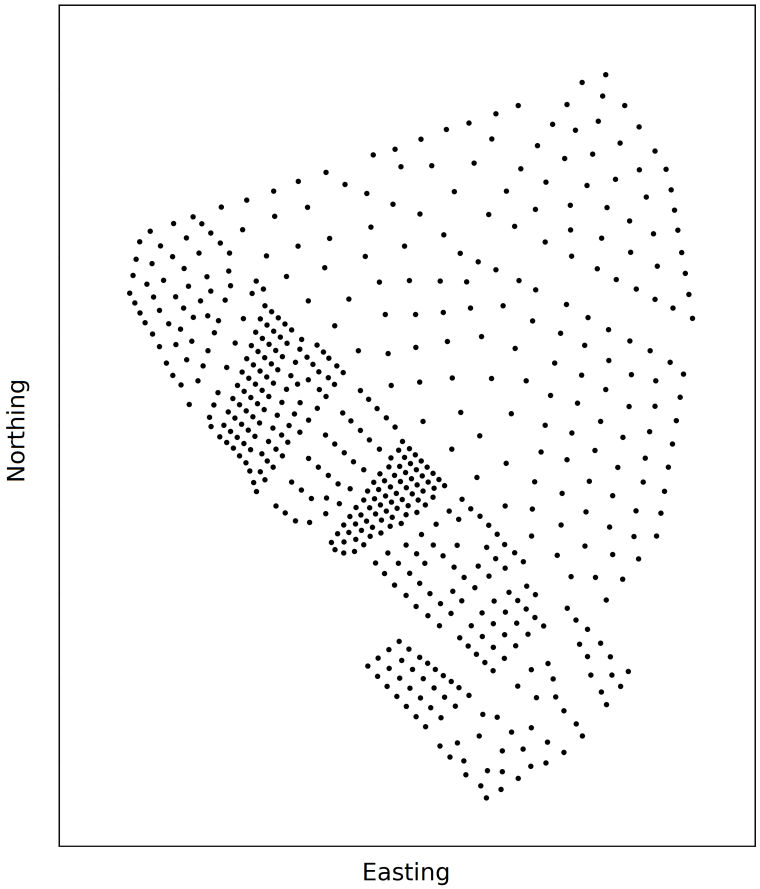


Figure 3.10: Wind farm layout considered for full zone test case.

lies in the fact that there is no data available regarding the yaw control implemented by all the turbines under consideration. In addition, even if we had this data, setting the yaw angles for each turbine at each time step would require a significant effort that would probably not be worthwhile.

The wake deficit model used is the GCH whose features are comprehensively described in Subsection 2.1.2. Although CC model may be the optimal model for this test case as it usually recommended for large farm applications, it is relatively new and demands greater computational resources compared to the GCH.

The input parameter values used can be found in Table 3.11. The selection of these values, which corresponds to the default ones used with analytical models, is due to a lack of information about the atmospheric stability and the turbulence intensity during the low pressure event. However, we are aware that these values should've been properly optimized for the current test case. The choice of $\alpha = 0.12$ implies selecting an unstable stratification condition consistent with typical values used for offshore applications [42, 43]; what could certainly have a more negative impact on the results is the selection of a turbulence intensity equal to 6%, certainly too small for the extreme conditions of the event. Thus, we are aware that these discrepancies will certainly negatively influence the comparison with the real data.

Wind direction and speed data used as input in UFLORIS are shown in Figure

Wind field		**Wake expansion**		**Added turbulence**			
I_0	α_s	k_a	k_b	$k_{f,a}$	$k_{f,b}$	$k_{f,c}$	$k_{f,d}$
0.06 [9]	0.12 [9]	0.38 [9, 11]	0.004 [9, 11]	0.5 [9]	0.8 [9]	0.1 [9]	-0.32 [9]

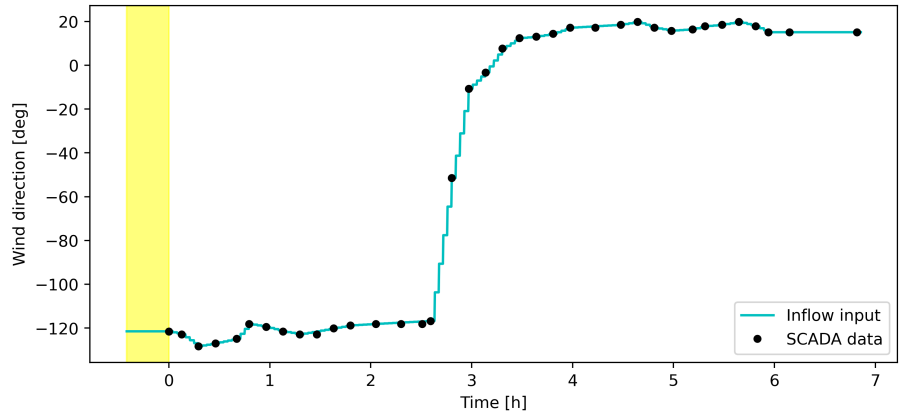
Potential core		**Power**	
α	β	η	p_p
0.58 [9, 23]	0.077 [9, 23]	1 [9]	1.8 [9, 38]

Table 3.11: GCH model parameter values used in UFLORIS, with references.

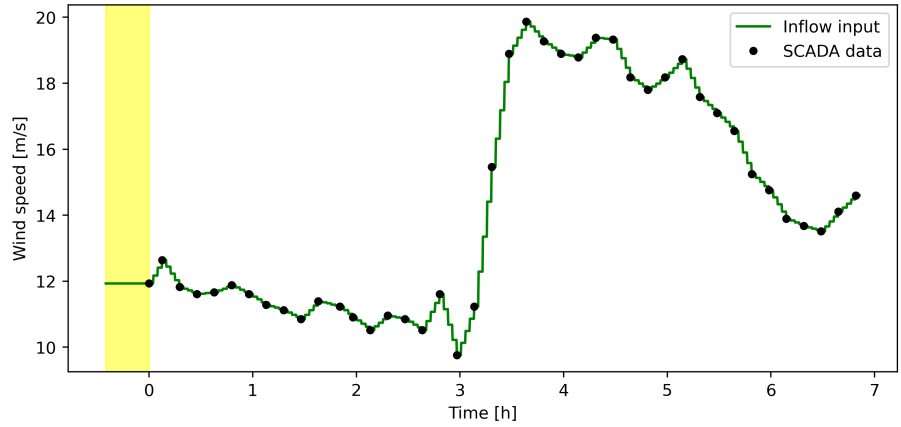
3.11. These input values have been derived from the SCADA data reported in [59], linearly interpolated every 150.6 s to assure a larger time resolution. For the sake of clarity of the figure, in yellow is depicted the initialization time and the time at which the unsteady computation starts is set at $t = 0$ s.

3.3.5 Post-processing of the results

The results obtained within the UFLORIS framework are post-processed to align them with the actual wind turbine control settings derived from SCADA data. Indeed, the analysis of the real data shows that during the fast changes in wind direction and speed around the third hour of the event, mostly of turbines are turned off to prevent serious damage during the extreme event. Additionally, there is a lack of data related to one wind turbine and another turbine remains turned off throughout the event. For these reasons the power outcome of each wind turbine is modified, setting the power equal to zero during the respective turning-off phase and neglecting the results of the missing wind turbine.



(a)



(b)

Figure 3.11: Wind direction (a) and speed (b) input data used for the low pressure system full zone test case. The initialization time is indicated in yellow; the unsteady computation starts at $t = 0$ s.

Chapter 4

Results

4.1 Comparison of the dynamic wake models

Here we present the results of the intercomparison between low- and medium-fidelity solvers, focusing on accuracy and runtime analysis for a standard configuration involving three wind turbines in a row with time-varying yaw control on the upstream turbine. The objective is to highlight differences in dynamic wake modeling and their impact on power output computations.

Initially, we assess the low-fidelity solvers, also including the quasi-steady mode, to outline the strengths and limitations of UFLORIS and FLORIDyn in capturing dynamic wake features. We emphasize the differences observed when these solvers are applied to scenarios with a steady background flow. Subsequently, we compare these low-fidelity solvers against the higher-fidelity FAST.Farm, which incorporates a turbulent inflow simulation. This comparison aims to assess the level of agreement with a more accurate tool, to explore how the different convective velocities of the OPs in the low-fidelity solvers affect the results, and to analyse the computational cost of the different wake modelling frameworks.

4.1.1 Low-fidelity models

Below are presented the results obtained with the low-fidelity simulators, where a steady inflow is considered. Since the objective of the comparison is to analyze the differences between using a stationary wake model, such as FLORIS, and dynamic solvers like FLORIDyn and UFLORIS, the initialization phase is omitted with $t = 0$ indicating the instant when the upstream turbine starts yawing.

4.1.1.1 Upstream wind turbine

In Figure 4.1, the trends of the upstream turbine power over time obtained using the three different solvers are shown, with an indication of yaw control. Using a notation similar to that employed in [12], the set of steady-state computations in FLORIS is denoted as Quasi-Steady (QS) FLORIS simulation.

The three wake models exhibit perfect agreement in the upstream turbine power computation. This alignment is due to the consideration of a steady inflow, causing power to instantaneously adjust to its control settings: as the yaw angle increases,

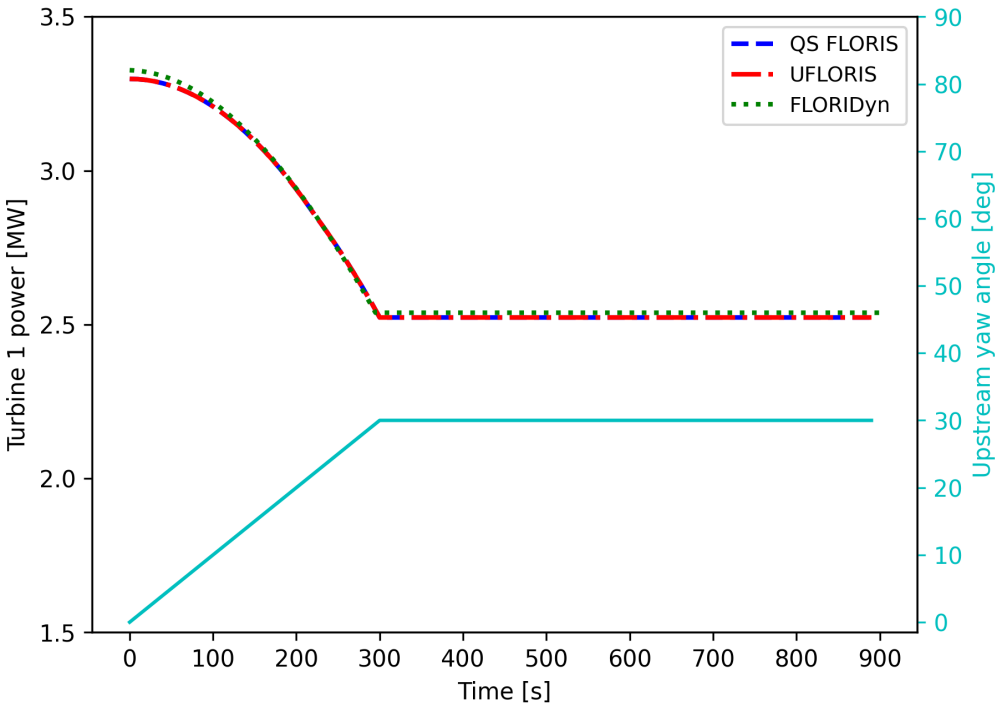


Figure 4.1: Upstream turbine power computed with the different low-fidelity wake models.

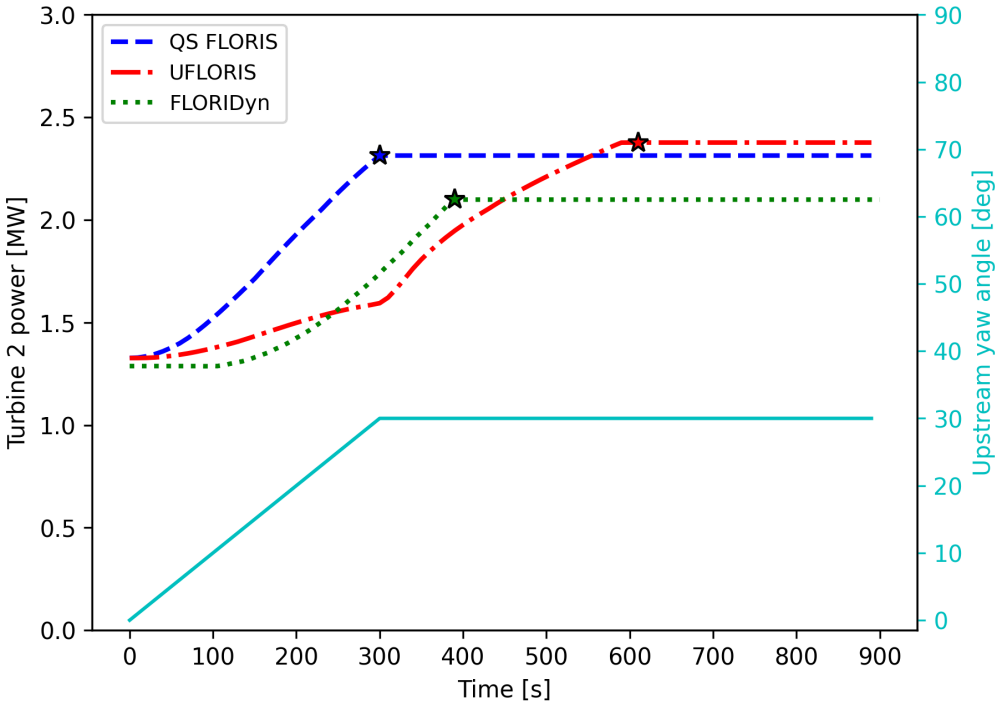
power decreases accordingly, reaching a constant value while the yaw angle is maintained, after $t = 300$ s. In addition to this, the accordance between the two unsteady solvers is due to the selection of a 2D horizontal distribution mode of OPs in FLORIDyn, closer to the approach used in UFLORIS, and the adjustment of the model parameter values (see Table 3.11).

4.1.1.2 Wake advection effects on the downstream turbines

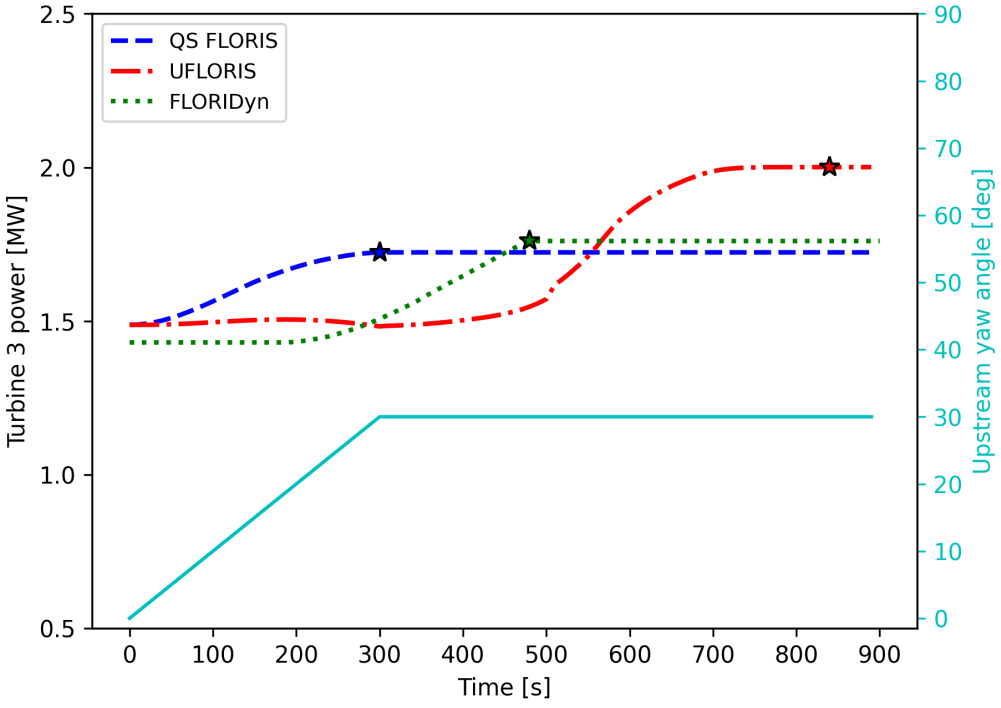
While it is true that increasing the yaw error of turbine 1 results in a decrease in the power collected from it, on the other hand, due to the redirection of the wake away from the downstream turbines, in this simplified case of three turbines in a row, one would expect an increase in power production from the downstream turbines and indeed, the results obtained from the three solvers confirm this, as shown Figure 4.2.

However, there are significant differences regarding the way the wake and thus the downstream turbine power is computed in three low-fidelity models. A notable discrepancy is observed when comparing the QS FLORIS simulation with those conducted using UFLORIS and FLORIDyn: the power calculated with the steady solver instantaneously changes according to the yaw condition of the upstream turbine, confirming that FLORIS does not capture the dynamic effects of the wake. Thus, at $t = 300$ s, when the yaw angle of the upstream turbine reaches its maximum value of 30 degrees, the wake of the upstream turbine deflects accordingly, leading to an increase in the power of turbines 2 and 3.

Instead, the results in UFLORIS and FLORIDyn show a delay between the



(a)



(b)

Figure 4.2: Downstream turbines (*Top:* turbine 2, *Bottom:* turbine 3) power computed with the different low-fidelity wake models. The stars indicate the time at which the steady condition is reached in the different solvers.

Wake model	$\boldsymbol{\Delta_{wt2}}$ $[\boldsymbol{s}]$	$\boldsymbol{\Delta_{wt3}}$ $[\boldsymbol{s}]$
FLORIS	0.0	0.0
UFLORIS	310.0	540.0
FLORIDyn	90.0	180.0

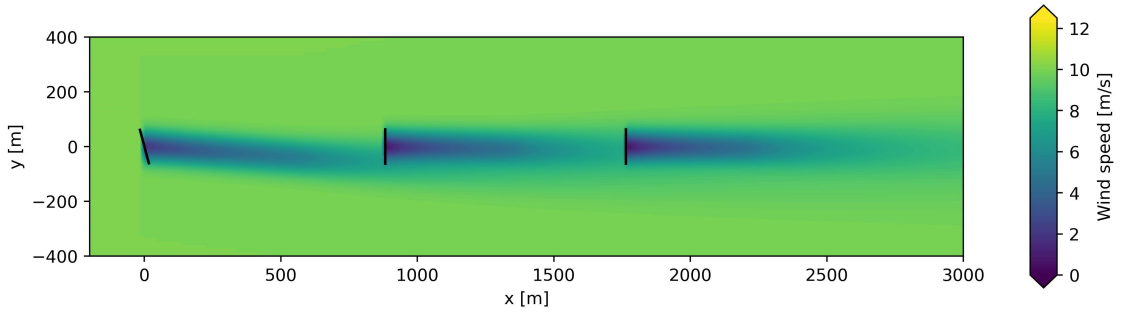
Table 4.1: Wake advection delay in reaching the steady condition for the two downstream wind turbines using the different low-fidelity wake models.

change in the upstream turbine yaw angle and its impact on the downstream turbine power generation due to wake advection. To clarify the differences in terms of wake advection among the three models, Figure 4.2 illustrates with a star the point at which the steady condition is reached for each wake model, i.e. the information of the maximum yaw angle arrives at the downstream turbine rotor so that its power remains constant over time. In other words, for these low-fidelity solvers, the steady condition is reached when the power maintains the same value between two consecutive time steps.

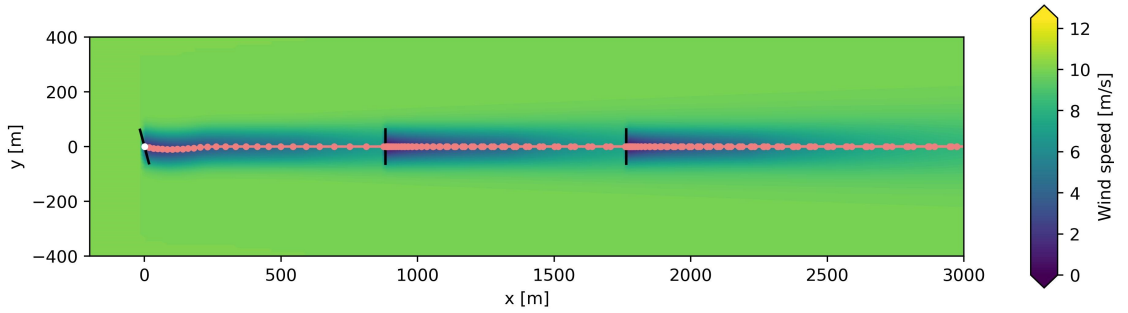
Additionally, in Table 4.1, the delay of the i-th turbine, Δ_{wti}, is reported, calculated as the difference between the time at which the power reaches the steady condition and the time at which the maximum yaw angle is reached ($t = 300$ s). Clearly, in the QS FLORIS simulation the dynamic nature of the wake is neglected, resulting in $\Delta_{wt2} = \Delta_{wt3} = 0$; conversely, the two unsteady solvers capture the wake advection delay. The differences in wake advection between the two unsteady solvers can be attributed to the different way the Observation Points (OPs) are transported through the flow field. Indeed, OPs play a crucial role in capturing the unsteady features of the wake, as they store information used to compute the velocity deficit during their propagation in the domain. Regarding the different Δ_{wti} values, the downstream turbine reaches a steady condition when the OPs storing the updated turbine state of $\gamma = 30$ deg reach its rotor. The longer time delay observed in UFLORIS is due to the fact that OPs in this solver travel at the local wind speed, while in FLORIDyn, they move at the free stream velocity, which is 10 m/s. As a result of the low-velocity region behind the upstream turbine, OPs generated in UFLORIS are transported at a slower speed in the near wake region and accelerate as the wake recovers. Nevertheless, local velocities consistently remain lower than the inflow speed, leading to a greater delay in wake advection computed in UFLORIS compared to the one obtained in FLORIDyn.

The visualizations of the flow field can help to better understand the wake unsteady features captured by the two dynamic wake models and the differences in the wake shape. In particular, in Figure 4.3 and 4.4 the contours of the velocity field computed at the hub height with the three solvers at two different time steps, $t = 160$ s and $t = 350$ s, are shown.

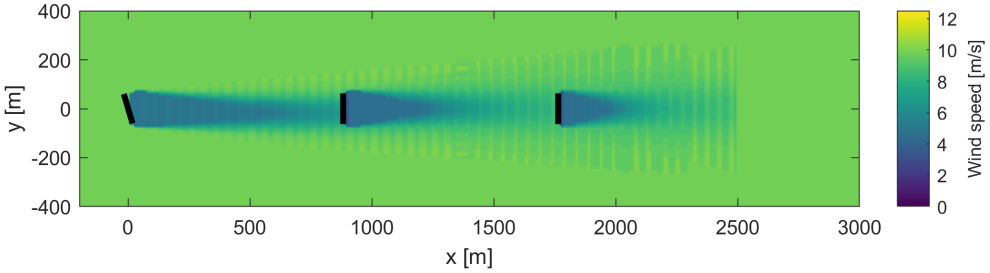
First, the focus is on comparing FLORIS and UFLORIS simulations. For the latter, a visualization of the OPs defining the wake centerline of the three turbines is provided; for a clear understanding of the wake advection effects, the OP generated at $t = 160$ s at the upstream turbine rotor is depicted in white. As shown in



(a) FLORIS

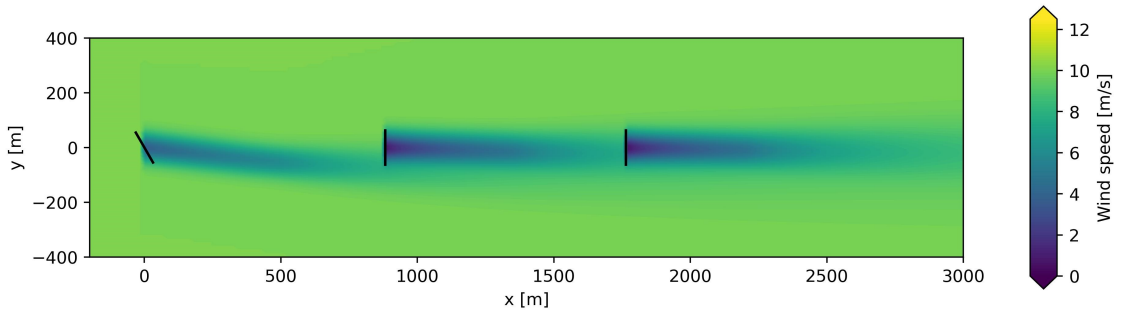


(b) UFLORIS

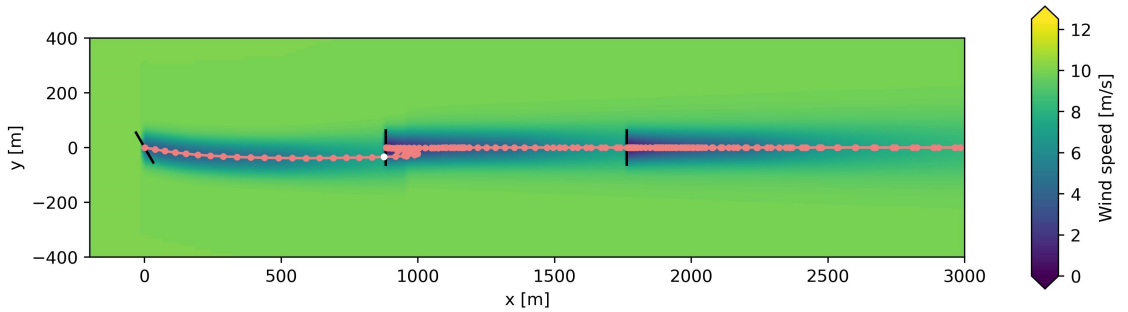


(c) FLORIDyn

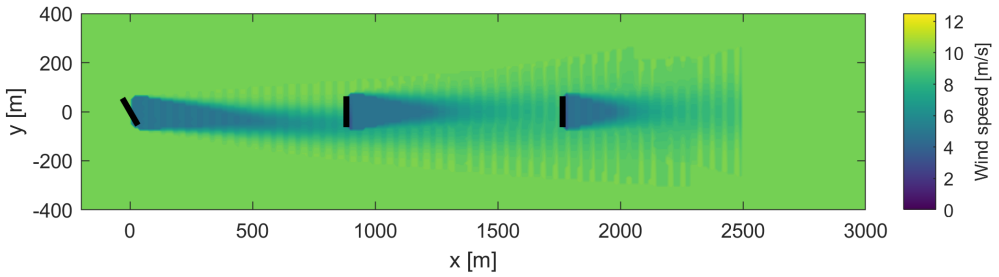
Figure 4.3: Contours of the wind speed at the turbine height at $t = 160$ s obtained with the different low-fidelity wake models.



(a) FLORIS



(b) UFLORIS



(c) FLORIDyn

Figure 4.4: Contours of the wind speed at the turbine height at $t = 350$ s obtained with the different low-fidelity wake models.

Wake model	Power [MW]		Percent difference [%]	
	wt2	wt3	wt2	wt3
FLORIS	2.31	1.72	/	/
UFLORIS	2.37	2.00	+2.73	+16.1
FLORIDyn	2.10	1.76	-9.28	+2.13

Table 4.2: Power output computed by the different solvers in the steady condition with the upstream turbine yawed of 30 degree.

Figure 4.3, when the turbine is yawed, the upstream wake centerline computed in UFLORIS intersects the downstream turbine rotor in the middle; in FLORIS, on the other hand, the wake is fully deflected according to the current turbine setting and intersects the rotor of turbine 2 on the right side (looking downwind). As illustrated in Figure 4.4.b, at $t = 350$ s, the OP generated at $t = 160$ s reaches the downstream turbine, and the upstream wake exhibits a shape that is not yet fully established since the information about reaching a steady condition has not yet been transported throughout the entire domain.

When considering the results obtained in FLORIDyn, it is observed that at $t = 160$ s (Figure 4.3.c), the upstream wake is slightly deflected towards the rotor of the downstream turbine (although not to the same extent as predicted by FLORIS) unlike what occurs in UFLORIS. Indeed, since the OPs are transported at 10 m/s, at $t = 160$ s, the wake deficit at the rotor of turbine 2 is calculated considering the yaw angle at time $t = 70$ s. At $t = 350$ s, the deflection of the wake from turbine 1 calculated with FLORIDyn is closer to the one predicted by FLORIS rather than by UFLORIS because we are closer to the steady condition (see Table 4.1).

Thus, it emerges that both UFLORIS and FLORIDyn allow to account for time delays between changes in the turbine states and their effects on the downstream wind turbines, even though these delays have different magnitudes because of the different OPs travelling modes in the two wake models.

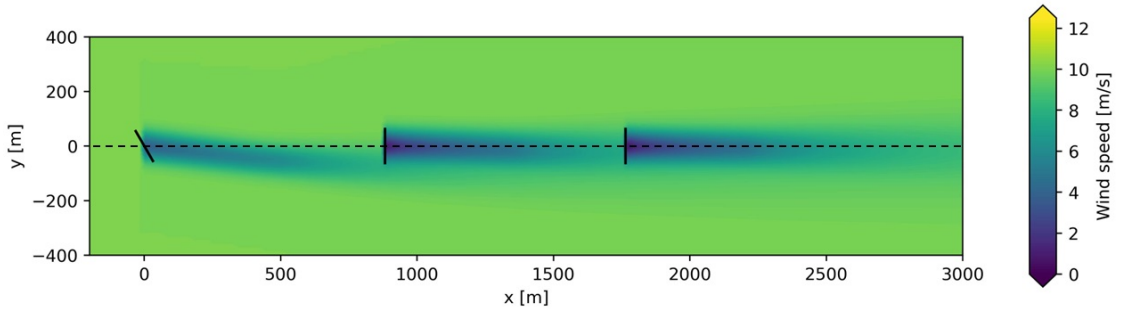
4.1.1.3 Differences in the steady condition

Regarding the steady power condition, illustrated in Figure 4.5, the three wake models predict different power outputs for the downstream wind turbines, as indicated in Table 4.2. It is also reported the percent difference in the power output using FLORIS results as baseline, which is defined as:

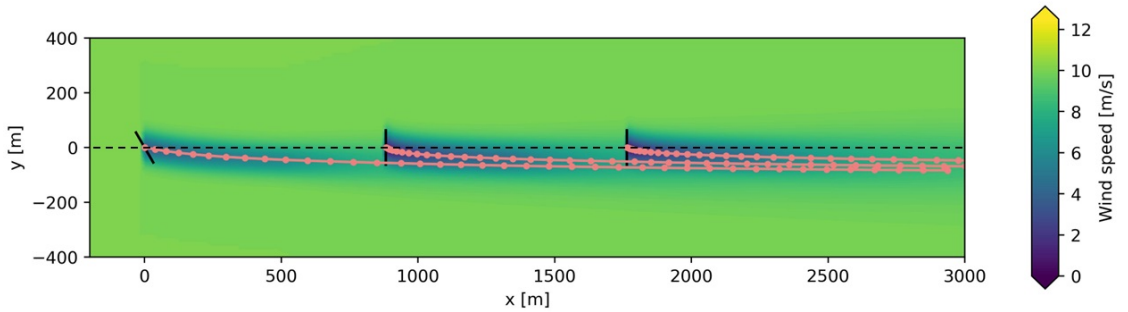
$$\text{Percent difference} = \frac{P - P_s}{P_s} \cdot 100 \tag{4.1}$$

where P_s is the mean power computed with FLORIS and P_u refers to the mean power obtained with one of the two unsteady models.

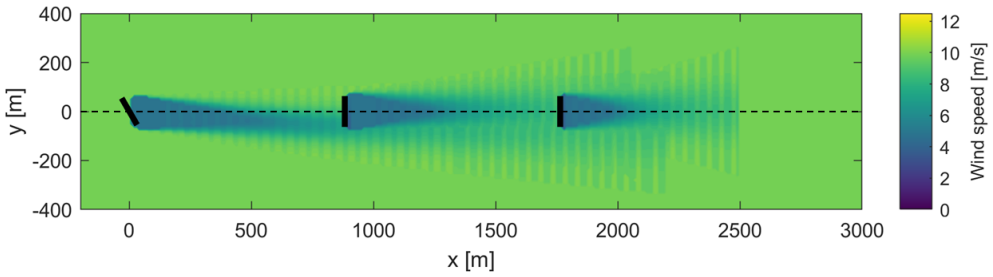
The differences between FLORIDyn and the other two solvers certainly lie in the different wake models used. Indeed, while FLORIDyn is based on the Gaussian model by Bastankhah and Porté-Agel [29], the GCH model [33] is employed in the UFLORIS and FLORIS simulations.



(a) FLORIS



(b) UFLORIS



(c) FLORIDyn

Figure 4.5: Contours of the wind speed at the turbine height at the steady condition obtained with the different low-fidelity wake models.

FLORIDyn is not able to capture the the secondary effects of wake recovery and secondary steering, which instead are included in the GCH model. As described in Subsection 2.1.2, secondary steering is the phenomenon where the counter-rotating vortices due to wake steering influence the downstream wakes inducing a deflection even if the downstream turbines are aligned with the wind direction. The added-yaw recovery allow to account for the faster recovery in misaligned conditions due to large eddies-induced entrainment in the wake field.

This clarifies why the wakes of downstream turbines computed with FLORIDyn align with the wind propagation direction, showing no steering, whereas in FLORIS, the wake centerlines of non-yawed turbines are slightly deflected because of the secondary steering effect.

The higher turbine 2 power output calculated with the steady solver is, therefore, attributable to the secondary effect of the enhanced wake recovery. However, this does not explain why FLORIDyn predicts an even higher power output for turbine 3, albeit slightly, compared to the steady solver. The reason for this seemingly counterintuitive behaviour lies in the different values of the added turbulence parameters (see Table 3.3). These parameters have a greater impact on the power of turbine 3 because, in the steady condition, the wakes of downstream turbines are fully merged. The use of the same parameters in FLORIDyn as those used for the GCH model would have resulted in a lower power output for the downstream turbine: the percent differences would have been -9.28% for turbine 2 and -17.5% for turbine 3. Thus, the lower percentage differences obtained using the values in Table 3.3 confirm the correct selection of the parameters for a fair comparison between FLORIDyn and FLORIS.

What arouses greater interest are the differences in the steady condition calculated by FLORIS and UFLORIS, both of which use the GCH model. Comparing the flowfield contours in Figure 4.5, it is evident that the cause of these differences lies in the different modeling of wake deflection. This is even more evident when visualizing the wake centerlines of downstream turbines in the near wake region, where an overestimation of wake lateral displacement is observed compared to the steady solver. Indeed, in UFLORIS the wake deflection is adjusted based on the OPs location which defines the unsteady wake centerline. Since OPs are transported with the local speed, also the transversal wake velocities, included in the GCH model, influence the OPs location. If in both FLORIS and FLORIDyn the deflection is based on the Gaussian deflection model, as described in Subsubsection 2.2.2.5, UFLORIS compute the lateral displacement of the wake based on the curl deflection model [66]. A visualization of the different wake deflection models is provided in Figure 4.6 by Martínez-Tossas et al. [66]; in addition to the aforementioned wake models, the lateral displacement of the wake centerline for LESs and the model from Shapiro, Gayme, and Meneveau [67] are also included.

For major details about the transversal wake velocities responsible of the disagreement in the wake deflection between UFLORIS and FLORIS, the reader is referred to Subsubsection 2.1.2.3.

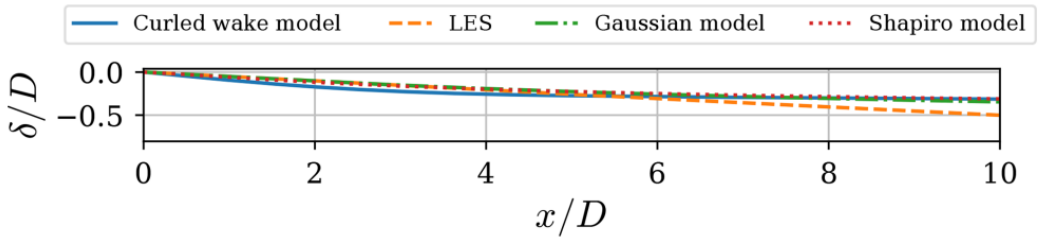


Figure 4.6: Lateral displacement of the wake centerline along the spanwise direction, both nondimensionalized with the turbine diameter, for different wake deflection models [66].

4.1.1.4 Issues in UFLORIS

Regarding the UFLORIS simulation, the power trend during wake advection remains unclear. The problem arises from the sudden change in the power output of the downstream turbines, which instead is not predicted in the FLORIDyn results, as shown in Figure 4.2. Notably, in the UFLORIS results, the power begins to fluctuate from the onset of yaw control, at $t = 0$ s. Analogous to FLORIDyn, one would expect a change in power output only after the upstream OP chain reaches the rotors of turbines 2 and 3. This issue is not yet fully understood and remains unsolved. One potential explanation could be the lack of transport of turbulence intensity with the OPs: when the upstream turbine starts yawing the change in the turbulence would be instantaneously convected in the full domain, influencing the power of downstream turbines.

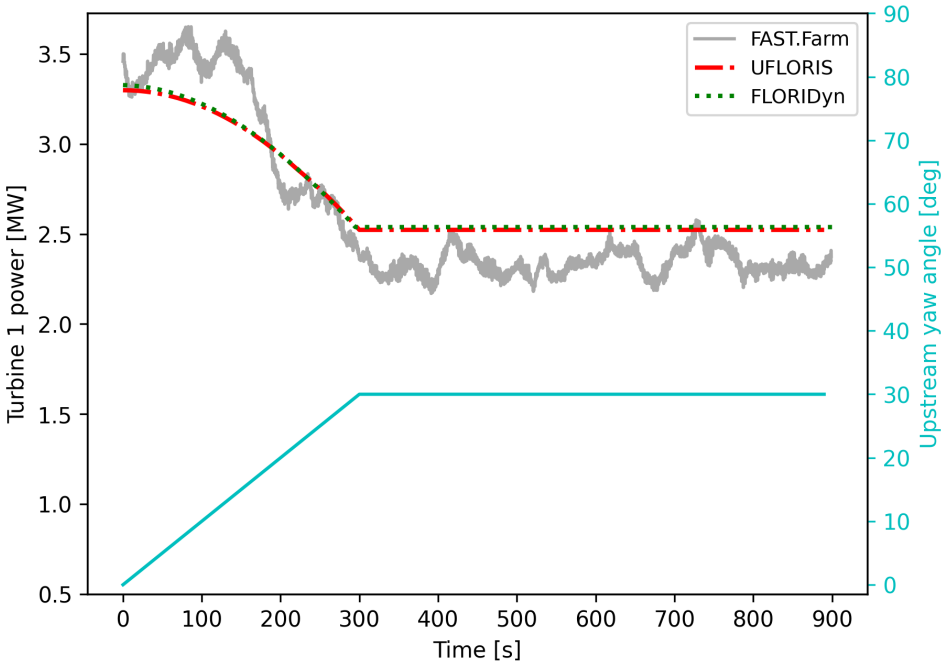
4.1.2 Comparison with FAST.Farm

Here we illustrate the comparison of the low-fidelity dynamic wake models, UFLORIS and FLORIDyn, with the medium-fidelity wake modeling tool FAST.Farm. This comparison aims to highlight the capabilities and limitations of the engineering models concerning wake modeling and power computation, as well as to validate the results using a higher fidelity solver.

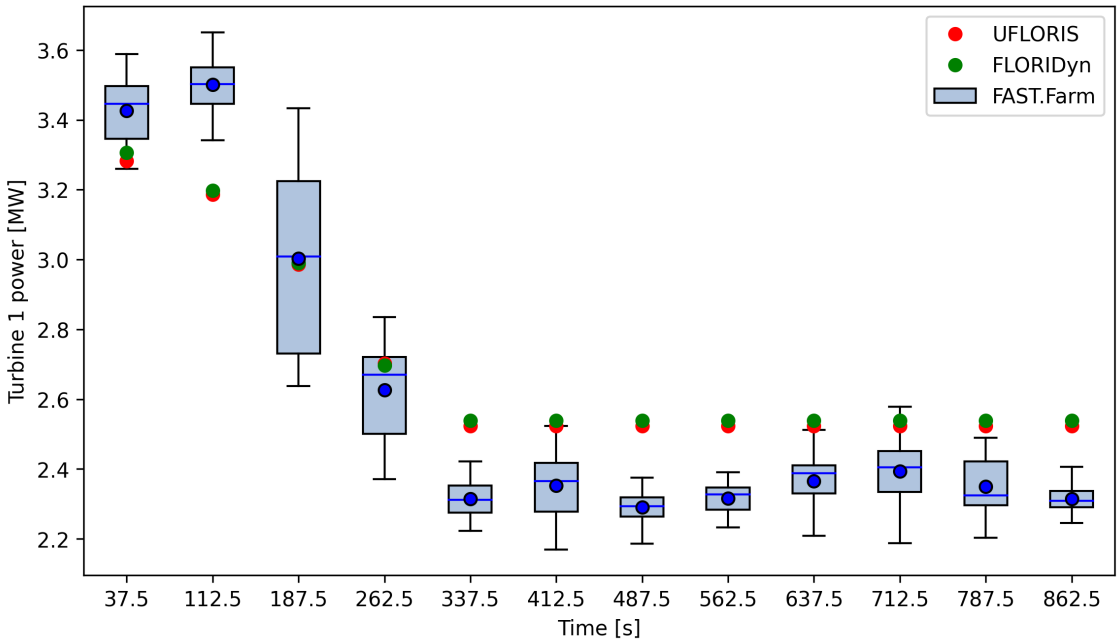
4.1.2.1 Power output

The focus is on the results obtained starting from the onset of yaw control of the upstream turbine. Therefore, the first 300 s, which are necessary to initialize the wake field across the computational domain in FAST.Farm, are excluded. The results obtained with the medium-fidelity solver are derived by averaging data from the twelve simulations using different turbulent inflows generated in TurbSim. Further details are provided in Subsection 3.1.3.

In Figure 4.7, 4.8, and 4.9, we compare the power generated by the three wind turbines in the engineering models with the results from FAST.Farm. Alongside the time-series data of the power output, we also present binned results to provide a clearer visualization of the differences and facilitate a better comparison between the wake models. In particular, the results are binned every 75 s. The choice of

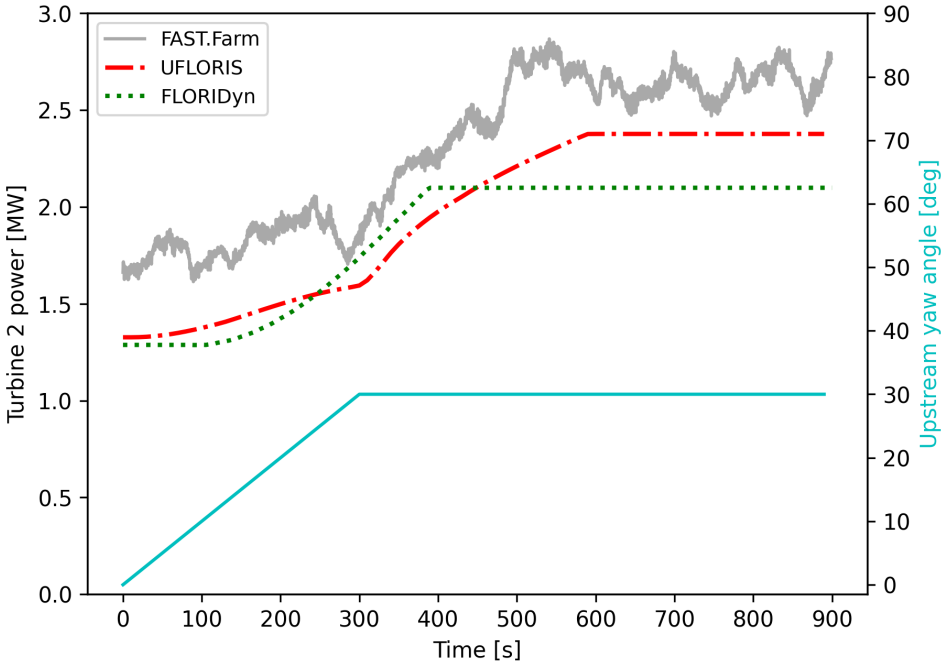


(a) Time-series of the power output with visualization of the upstream turbine yaw control; the power output from FAST.Farm is averaged across different inflow conditions generated in TurbSim.

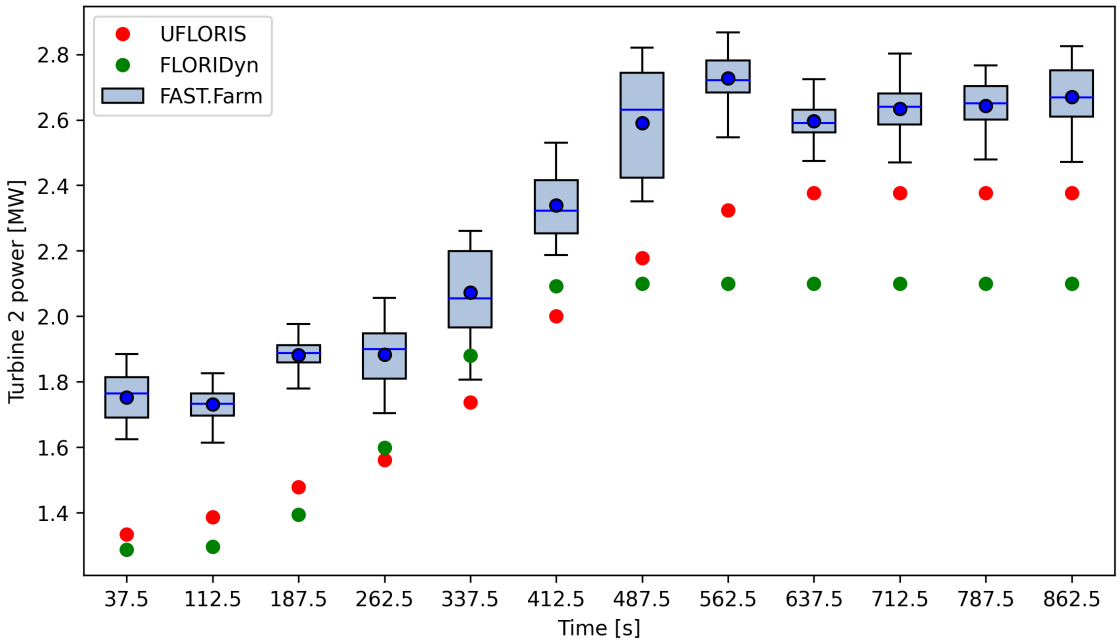


(b) Boxplot of the power output, binned every 75 seconds, computed using simulations in FAST.Farm; the plot includes a comparison with mean power values computed using UFLORIS and FLORIDyn within the same time bin.

Figure 4.7: Upstream turbine power computed with the low-fidelity unsteady wake tools and FAST.Farm.

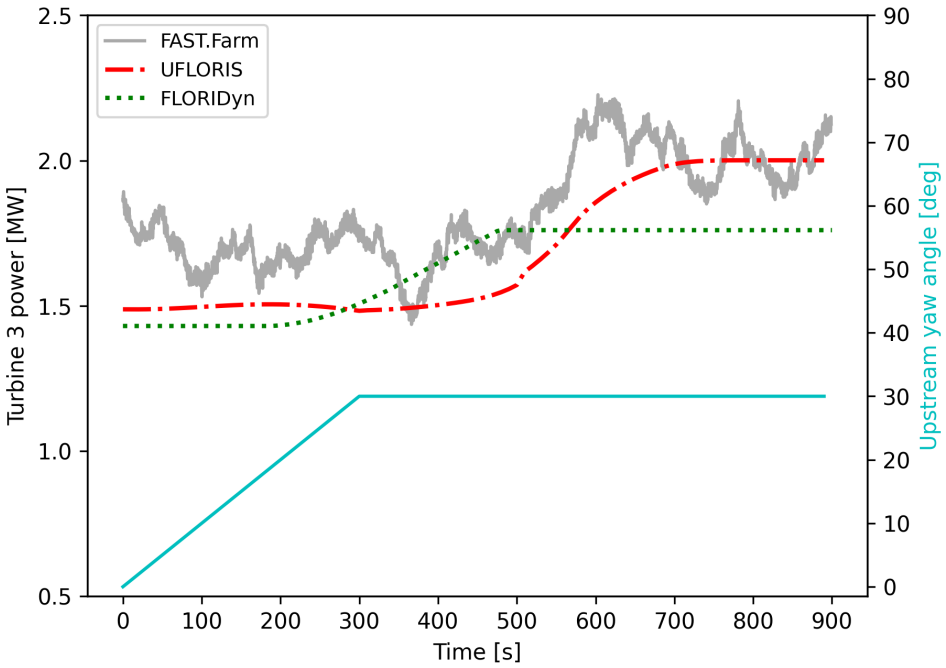


(a) Time-series of the power output with visualization of the upstream turbine yaw control; the power output from FAST.Farm is averaged across different inflow conditions generated in TurbSim.

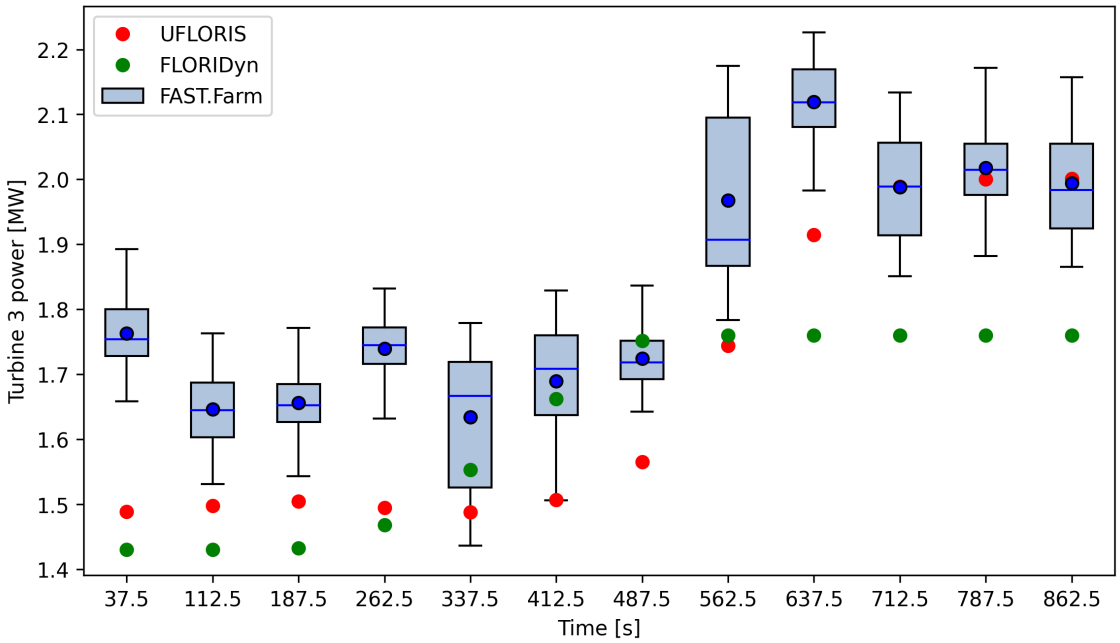


(b) Boxplot of the power output, binned every 75 seconds, computed using simulations in FAST.Farm; the plot includes a comparison with mean power values computed using UFLORIS and FLORIDyn within the same time bin.

Figure 4.8: Turbine 2 power computed with the low-fidelity unsteady wake tools and FAST.Farm.



(a) Time-series of the power output with visualization of the upstream turbine yaw control; the power output from FAST.Farm is averaged across different inflow conditions generated in TurbSim.



(b) Boxplot of the power output, binned every 75 seconds, computed using simulations in FAST.Farm; the plot includes a comparison with mean power values computed using UFLORIS and FLORIDyn within the same time bin.

Figure 4.9: Turbine 3 power computed with the low-fidelity unsteady wake tools and FAST.Farm.

Time bin center [s]	Power error [MW]					
	Turbine 1		Turbine 2		Turbine 3	
	UFLORIS	FLORIDyn	UFLORIS	FLORIDyn	UFLORIS	FLORIDyn
37.5	-0.145	-0.120	-0.418	-0.464	-0.274	-0.333
112.5	-0.316	-0.304	-0.344	-0.434	-0.149	-0.216
187.5	-0.016	-0.012	-0.403	-0.487	-0.152	-0.224
262.5	0.076	0.071	-0.322	-0.285	-0.245	-0.271
337.5	0.209	0.224	-0.336	-0.193	-0.146	-0.081
412.5	0.170	0.186	-0.338	-0.246	-0.183	-0.028
487.5	0.232	0.248	-0.412	-0.491	-0.159	0.027
562.5	0.206	0.222	-0.404	-0.628	-0.224	-0.207
637.5	0.157	0.173	-0.220	-0.498	-0.205	-0.359
712.5	0.130	0.145	-0.257	-0.535	0.001	-0.228
787.5	0.173	0.189	-0.267	-0.545	-0.017	-0.257
862.5	0.209	0.224	-0.294	-0.572	0.007	-0.234

Table 4.3: Power error made by the low-fidelity dynamic wake modeling tools compared to the binned mean power computed by FAST.Farm.

this bin size is based on the fact that FLORIDyn is, as demonstrated in the previous comparison, the solver in which the wake advection requires less time since the OPs are transported at the free stream velocity (10 m/s). Considering the distance between the turbines, equal to $7D = 882$ m, an advection time of 88.2 s is estimated. Therefore, to account for the advection delay, a bin size smaller than the latter value should be considered. The bin size value of 75 s ensures clear observation of the average trend in power over time while avoiding excessive data points in the graphs that could lead to misunderstandings. In each time-bin we report the mean power output computed with UFLORIS and FLORIDyn and represented with a dot. Given the greater dispersion of the FAST.Farm data, we represent them with a box-and-whisker plot. The blue dot and line represent, respectively, the mean and the median (middle value in the data). The box represents the middle 50% of the data, known as the interquartile range (IQR), giving an indication of the spread of the central data values; the lower edge of the box is the first quartile (Q1 or 25th percentile), and the upper edge is the third quartile (Q3 or 75th percentile). The horizontal lines outside the box define the edges of the whiskers, which extend from the box to a distance equal to 1.5 times the IQR. In addition, table 4.3 shows the absolute error incurred by the low-fidelity solvers in calculating the turbine power generation compared to the simulations performed in FAST.Farm; this error is based on the mean power values within each bin.

In terms of the power output of the upstream turbine (Figure 4.7), FAST.Farm predicts slightly higher power levels both when the turbine is aligned with the mean flow at $t = 0$ s and when the upstream yaw angle is maintained at 30 degrees. As demonstrated in the previous subsection, both UFLORIS and FLORIDyn yield identical power generation results as the FLORIS model due to the absence of turbulence wind speed changes. Therefore, addressing the discrepancies between the low-fidelity solvers and FAST.Farm requires adjustments to the steady wake model. Specifically, fine-tuning the parameters η and p_p, which are used to calculate turbine power as defined in Equation 2.49, could potentially reconcile the differences in re-

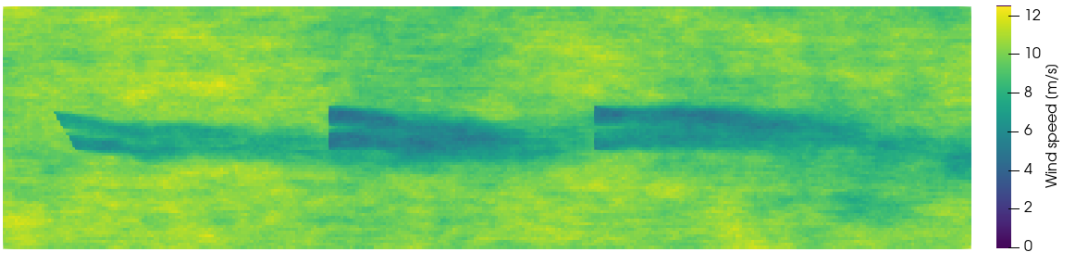
sults. However, this aspect has not been explored in this work as it falls outside the scope of the thesis. Nonetheless, the results demonstrate a good agreement, with mean power errors relative to FAST.Farm being less than 10%.

4.1.2.2 Dynamic response

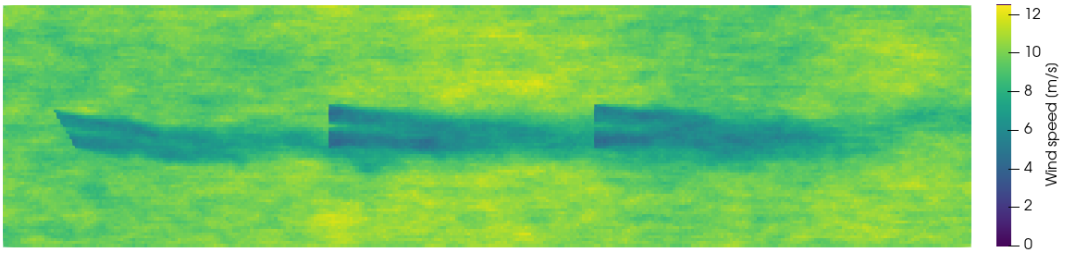
The results of the downstream turbine power provide insights into the dynamic wake modeling in the different wake models.

Focusing on the initial approximately 150-200 s, during which the upstream turbine rotor is misaligned from the mean wind direction with a yaw rate of 0.1 deg/s and where dynamic effects are not yet significant, both UFLORIS and FLORIDyn exhibit comparable errors. The magnitude of these errors can be attributed to the lack of tuning of the wake model parameters used. However, in the steady phase when the effects of a 30 deg yaw angle are transported to the position of turbine 3, UFLORIS shows better agreement with FAST.Farm results. It is deduced, therefore, that the inclusion of secondary steering and wake added yaw effects, considered in the GCH model used in UFLORIS but not in FLORIDyn, allows for more accurate results. Figure 4.10 depicts two screenshots of the two-dimensional flow field at hub height obtained in FAST.Farm in simulation I (refer to Table 3.9 for details on the random seeds used for turbulence generation), corresponding to the final phase of the simulation where FLORIDyn and UFLORIS have reached the steady condition. These flow fields can be compared with those predicted by the low-fidelity solvers and represented in Figure 4.5. Although the simulations are based on different inflows (turbulent in FAST and uniform in the engineering models), we find a better agreement between UFLORIS and FAST.Farm as a deflection is observed in the wake of downstream turbines. Instead, the Gaussian model used in FLORIDyn lacks in modeling the lateral displacement of downstream wakes due to upstream turbine yawing. This explains why there is a lower power error in the final phase of the UFLORIS simulation. However, from the comparison of the flow fields, it is also deduced that UFLORIS tends to overestimate the wake displacement of downstream turbines in the near wake zone, being more accurate only farther from the rotor.

It is of great interest to compare the magnitude of wake advection delay on downstream turbines. In the results obtained in FAST.Farm, it is not possible to precisely identify the moment when the effects of the steady upstream turbine yaw angle affect the downstream turbines, unlike the low-fidelity solvers. From plots in Figure 4.8 and 4.9, it is observed that the response of turbines 2 and 3 to the 30 deg yaw angle occurs within the ranges from 500 to 600 s and 700 to 800 s, respectively. Thus, a delay of the order of 200-300 s for turbine 2 and 400-500 s for turbine 3 is predicted in FAST.Farm. Looking at the results in Table 4.1, it emerges FLORIDyn underestimates and UFLORIS overestimates the wake advection delay in reaching the steady condition. This feature observed in FLORIDyn aligns with findings from [11], where Becker notes that the effects of state change occur too rapidly when OPs travel at the free stream velocity. On the contrary, UFLORIS exhibits a significantly larger delay because the OPs are convected at the local wind speed. Concerning the wake meandering characteristics in FAST.Farm, wake advects based on the local spatially averaged ambient wind speed and wake deficit. Figure 4.11 shows the comparison



(a) Time: $t = 750$ s.



(b) Time: $t = 850$ s.

Figure 4.10: Contours of the wind speed at the turbine hub height during the last 150 s of the FAST.Farm simuation; the turbulent inflow wind is the one generated in simulation I with TurbSim.

Wake model	Simulation CPU time
QS FLORIS	0.85 s
UFLORIS	11.4 s
FLORIDyn	41.4 s
FAST.Farm	30.89 min

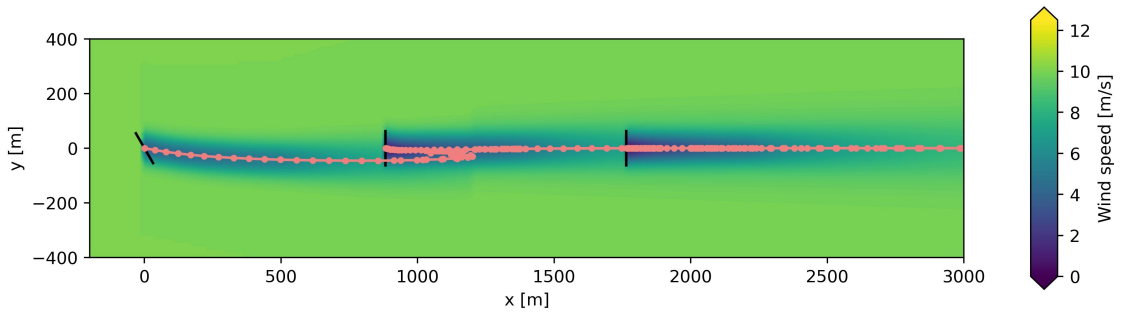
Table 4.4: Simulation CPU time for the different wake models; the computational time of FAST.Farm is averaged on the twelve simulations conducted.

of the flow field simulated with the three dynamic wake models at $t = 400$ s, when the upstream turbine wake deflection is fully established in FLORIDyn, and indeed the flow field is identical to the one computed at the last time step, shown in Figure 4.5.c. Instead in UFLORIS the upstream turbine wake deflection is different from the one computed and the end of simulation (see Figure 4.5.b). The comparison of the flow field in FAST.Farm at $t = 400$ s (Figure 4.11.a) and at the end of the simulation (Figure 4.10), although challenging due to the turbulent nature of the inflow, reveals that similar to UFLORIS, the wake response to the state change takes longer in FAST.Farm. This is evident as, on average at the end of the simulation, the wake tends to impact the right side of turbine 2 when viewed in the downstream direction.

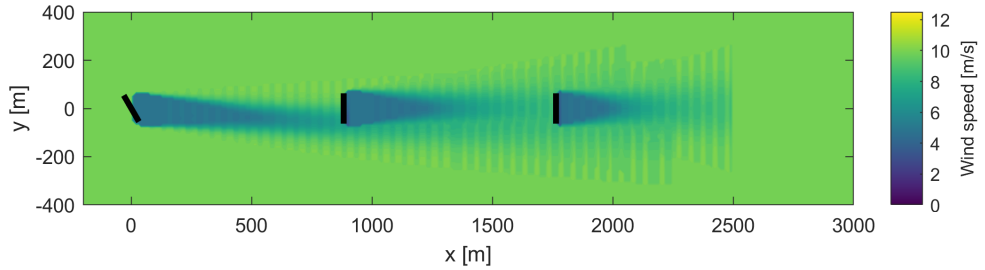
The optimal strategy for the convection of the OPs has not been definitively established in the literature. The intercomparison reveals that UFLORIS results show closer alignment with those obtained from FAST.Farm in terms of wake advection delay. However, further experiments with varying mean wind speeds are necessary to confirm or deny the stronger agreement of FAST.Farm with UFLORIS. What is certain is that to determine which approach is more accurate, it would be necessary to compare these three solvers with a higher-fidelity wake simulation tool, such as LES, which could assess the level of accuracy of the results.

4.1.2.3 Runtime

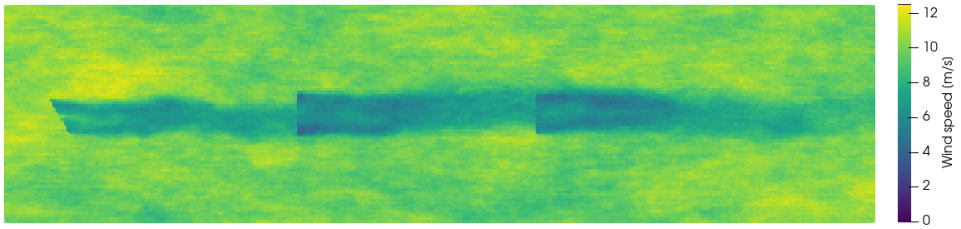
We report in Table 4.4 the computational time for the simulations done using the different wake modeling frameworks, also including the collection of steady-state computations. The results only meant to give an approximated estimation of the runtime performance of the different frameworks; these times exclude plotting. It should be noted that different workstations were used for the solvers: FLORIDyn and FAST.Farm results were obtained on a Windows 64-bit workstation with a 2.8 GHz 4-core Intel i7 CPU and 16GB of RAM; QS FLORIS and UFLORIS simulations were conducted on a Linux (Fedora) 64-bit workstation with 2 1.5 GHz 64-core AMD EPYC 7742 CPUs and 944.8 GB of RAM. It is emphasized that the simulations were not parallelized, so a single core was utilized. Other than the different devices used, which certainly influence the comparison, it should be noted that the run time is also influenced by the model settings such as initialization time (700 s for the low-fidelity solvers, 300 s for FAST.Farm) and the time step. Therefore, in this analysis, we look at the results considering the order of magnitude of the simulation CPU time.



(a) UFLORIS



(b) FLORIDyn



(c) FAST.Farm

Figure 4.11: Contours of the wind speed at the turbine height at $t = 400$ s obtained with the different dynamic wake models.

FLORIS has the lowest computational cost, requiring less than a second to simulate a total of 1600 s. However, due to the steady-state nature of the model, it does not account for dynamic effects, resulting in significant errors when considering time-varying turbine control settings, as in the simulated test case here, or when simulating unsteady wind conditions.

From the comparison of the run times of dynamic wake modeling frameworks, a significant advantage of the engineering wake models emerges. Indeed, while FAST.Farm requires an average[1] of 31 minutes to simulate 1200 s (thus 1.5 times slower than real-time), UFLORIS and FLORIDyn require a time on the order of tens of seconds for 1600 s of simulation (respectively, 160 and 40 times faster than real-time). Thus, for layouts with a limited number of turbines, these analytical wake modeling tools simulate faster than real time. The increase in computational time with the number of turbines has not been investigated; however, one would expect a consistently lower computational cost compared to the medium-fidelity solver. Although UFLORIS and FLORIDyn lack the fidelity of FAST.Farm, these engineering models show high potential in representing wake dynamic behaviours and are preferred in terms of computational time compared to higher fidelity tools.

4.2 UFLORIS development

In this section, we present the results obtained from the simple two-turbine test case described in Subsection 3.2.1. Firstly, we discuss the improvements achieved through modifications made to the UFLORIS source code. Secondly, we present the results obtained using the novel OPs deletion strategy, emphasizing the achievable computational time savings while maintaining good accuracy.

4.2.1 Implementation and improvements

To solve the issue related to the wake centerline computation when the unsteady approach is enabled, the novel version of UFLORIS neglects the influence of the ground, specifically its impact on the shed vortices modeled in the GCH wake model. As detailed extensively in the previous chapter, it is the ground mirror wake rotation vortex that primarily causes the transversal displacement of the OPs during the initialization time.

Figure 4.12 shows the snapshots of the flow field just before and after the start of the unsteady wake computation. The comparison with the corresponding figures in Figure 3.6 reveals that in the UFLORIS version without the ground effect:

- during the quasi-steady approach ($t \leq 200$ s) OPs accurately trace the real wake centerline calculated by FLORIS as a results of the zero transversal wake velocities at the location of the OPs;
- when the unsteady solver is enabled ($t = 200$ s) the wake centerline is adjusted based on the OPs locations and at $t = 210$ s there are no visible changes in

[1]The CPU time is obtained by averaging the twelve simulations done in FAST.Farm. Note that this time excludes the computational cost for generating the turbulent inflow with TurbSim, which is couple of hours for each time-series.

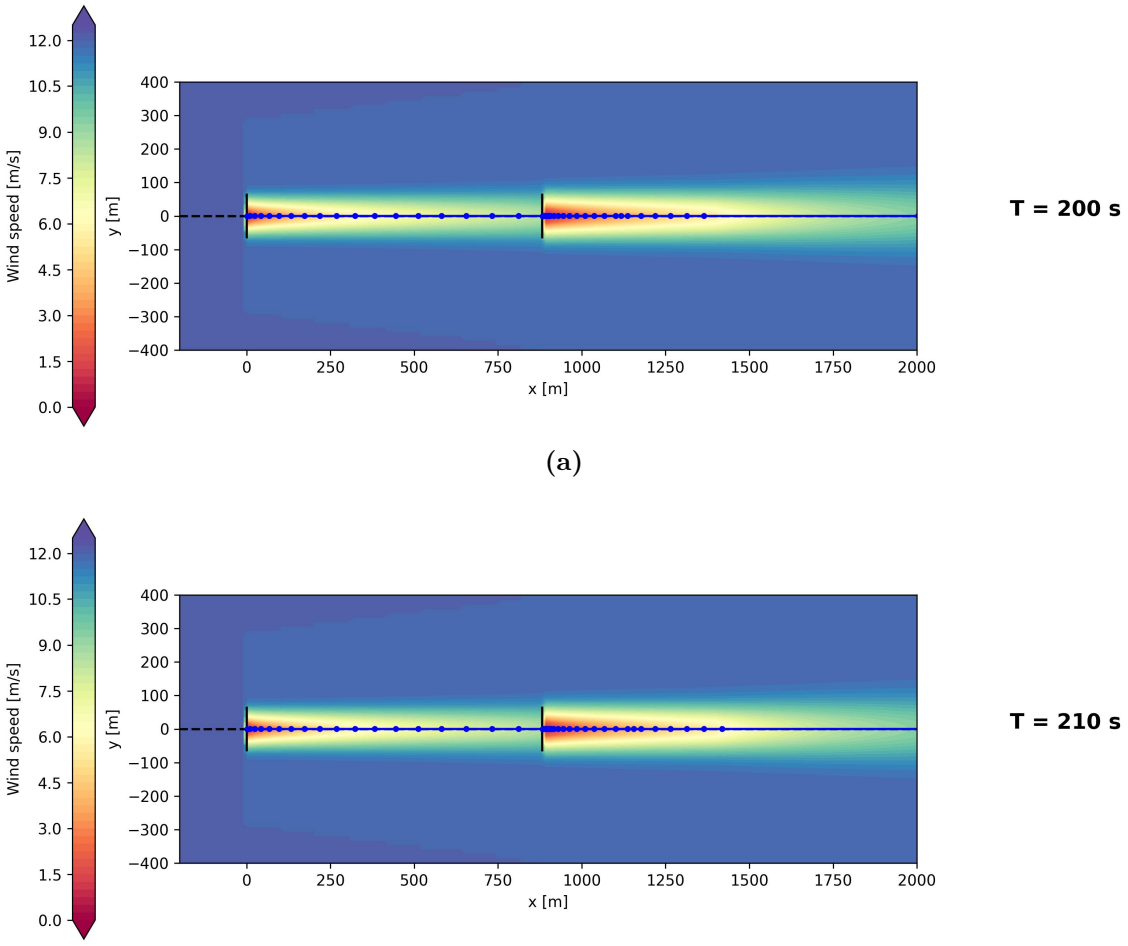
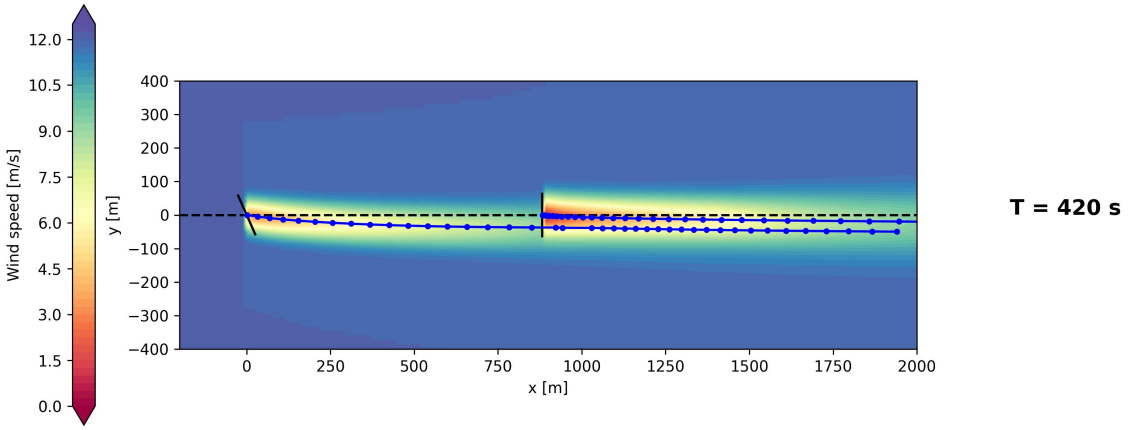


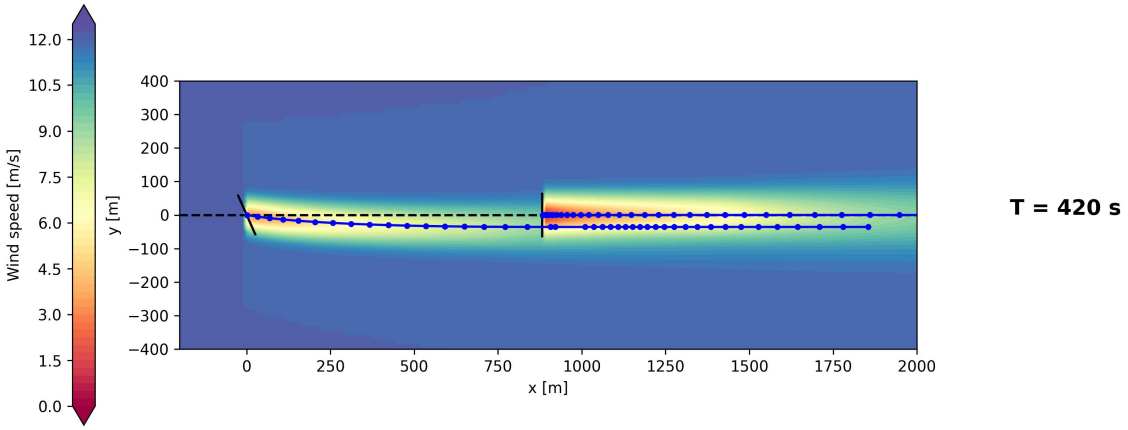
Figure 4.12: Contours of the wind speed at the turbine hub height with visualization of the unsteady wake centerline before (a) and after (b) the initialization time, obtained disabling the ground effect.

the wake shape because of the consistency between the steady and unsteady solvers at the previous time step.

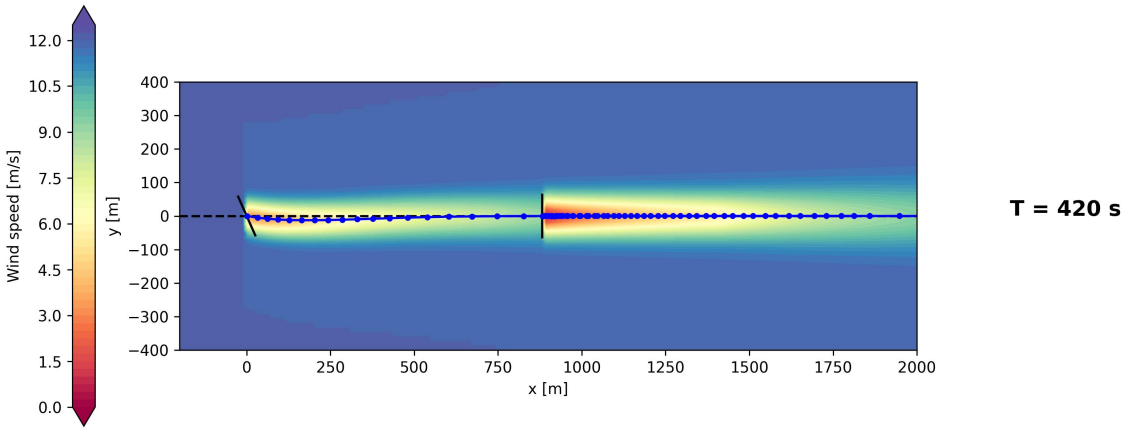
However, disabling the ground effect alone does not resolve the issue of the wrong wake deficit computation that is described in the previous chapter. To address this problem, we have modified how the thrust coefficient and yaw angle values transported by the OPs are used to calculate the wake field. Comprehensive details about these modifications are provided in Subsubsection 3.2.2.2. Evidence of UFLORIS improvements is shown in Figure 4.13, which illustrates the wind field contours at hub height at $t = 420$ s, during the unsteady approach. Particularly, Figure 4.13.c shows that the novel UFLORIS solver correctly captures the delay between the change in the upstream turbine yaw angle and its effects on the downstream turbine. Indeed, at $t = 420$ s, when the OPs with information about the yaw changes of the upstream turbine haven't reached the downstream turbine rotor yet, the upstream turbine wake show a consistent shape. Contrary, both the flow



(a) Before modifications.



(b) Without ground effect.



(c) After modifications.

Figure 4.13: Contours of the wind speed at the turbine hub height with visualization of the unsteady wake centerline at $t = 420$ s.

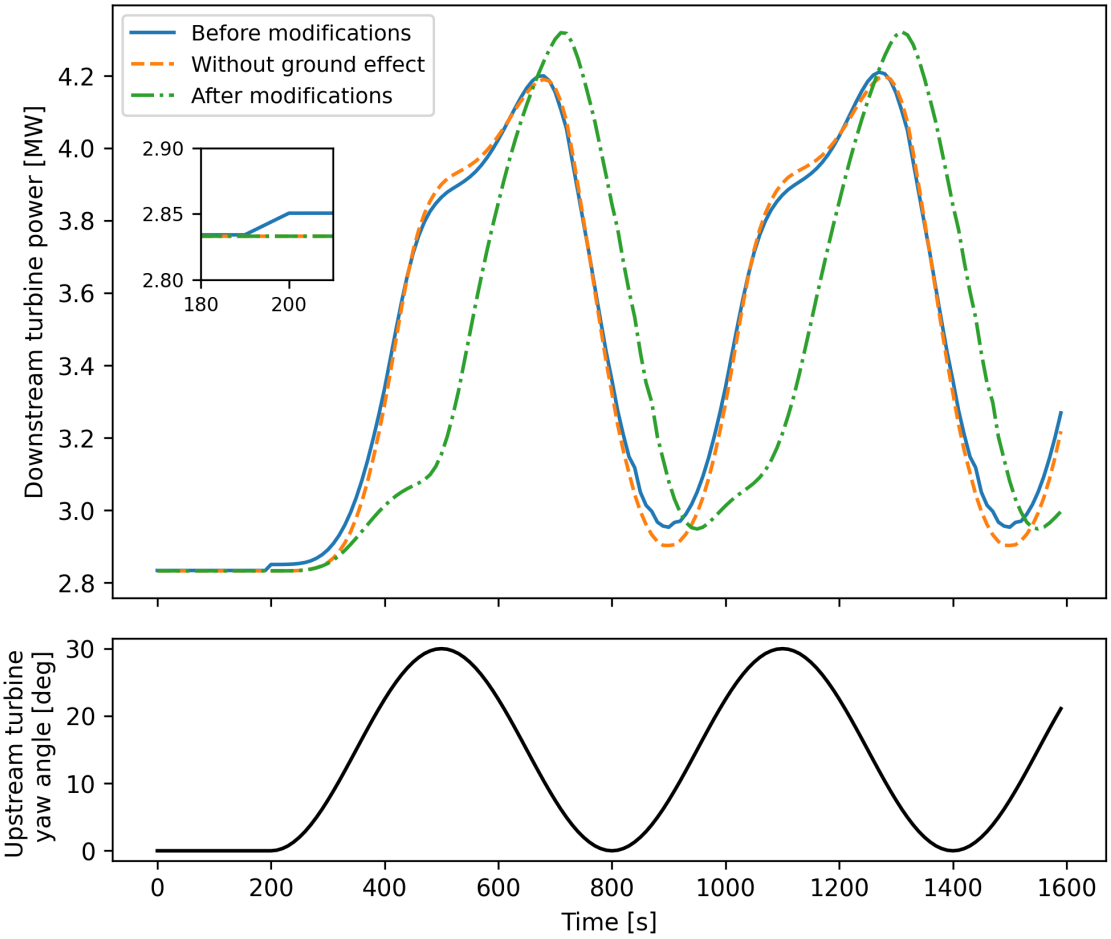


Figure 4.14: Comparison of the downstream turbine power output obtained with the original UFLORIS (before modifications), the version in which the ground mirror vortices are neglected (without ground effect) and the novel UFLORIS version (after modifications).

fields computed with the original version of UFLORIS and the version without the ground effect show changes in the entire upstream turbine wake, thus incorrectly capturing the advection delay.

The impact of the source code modifications on the computation of the downstream turbine power is illustrated in Figure 4.14; it displays the power variations resulting from the upstream turbine yaw control using three versions of the UFLORIS solver: the original version, the version neglecting the ground effect, and the newly implemented version. The comparison shows that:

- the modifications made to the definition of total transversal wake velocities (see Equation 3.15 and 3.16) lead to consistent power output just before and after the start of the unsteady approach. In contrast, the original UFLORIS solver predicts a sudden increase in the downstream turbine power at $t = 200$ s due to the differences between the position of the OPs and the wake centerline in the initialization phase;

- in the novel version of the UFLORIS solver the wave-shape of the downstream

power output shows notably differences with the version without the modification of the interpolation function for the thrust coefficient and the yaw angle values. All three versions of the unsteady solver predict a phase shift with respect to the prescribed yaw control, but in the modified version, the delay in the convection of the upstream turbine yaw angle values is greater, and furthermore, the maximum power output predicted in the unsteady phase is higher. The most significant difference, however, is observed in the initial phase of power output increase, which occurs periodically in the time range between 300 and 600 s. The versions "before modifications" and "without ground effect", indeed, predict much higher power due to the incorrect interpolation of C_T and γ, resulting in inaccurate modeling of the upstream turbine wake (see Figure 4.13).

To sum up, the modifications made allow to solve some of the issues in the UFLORIS solver: the removal of the calculation of transversal velocities induced by mirrored vortices to respect the ground boundary condition enables a calculation of the convection of OPs during the initialization time with major agreement to the steady solver; changes in the source code allow for a more accurate calculation of the wake velocity deficit consistently with the values of C_T and γ carried by the OPs. Overall, the novel UFLORIS solver ensures more physically accurate modeling of the wake.

However, further modifications are still necessary. In particular, the behavior of the power output of downstream turbines under time-varying yaw control conditions is not yet clear. For example, concerning the specific case, the initial phase of power growth that occurs periodically starting from the time range between 300 and 500 s is not explained. Instead, one would expect a power wave-shape consistent with the harmonic trend of the upstream turbine yaw control. One possible explanation is the lack of advection of the turbulence intensity with the OPs.

4.2.2 OPs deletion strategy

Here, we present the results obtained from implementing the OPs deletion strategy described in Subsection 3.2.3, which is based on imposing the number of OPs for each turbine chain. We compare this strategy with the default approach, which involves removing OPs when they are convected outside of the domain. We use the results of the default strategy as a baseline for comparison since the fixed number of OPs is such that they are fewer than those expected from the default strategy; thus, this assures greater accuracy in the dynamic wake modeling. As described below, the new strategy allows for a reduction in computational cost for the simulation while maintaining good accuracy by selecting an appropriate number of OPs.

Figure 4.15 shows the comparison of the results obtained in terms of downstream turbine power output. Results for the upstream turbine are not reported since the first turbine is not impacted by the wake of other turbines and always experiences steady wind, thus its performance is not influenced by the OPs deletion strategy. Additionally, the percentage error relative to the baseline results is reported only for simulations where a non-zero error was obtained. Evidently, having an equal or

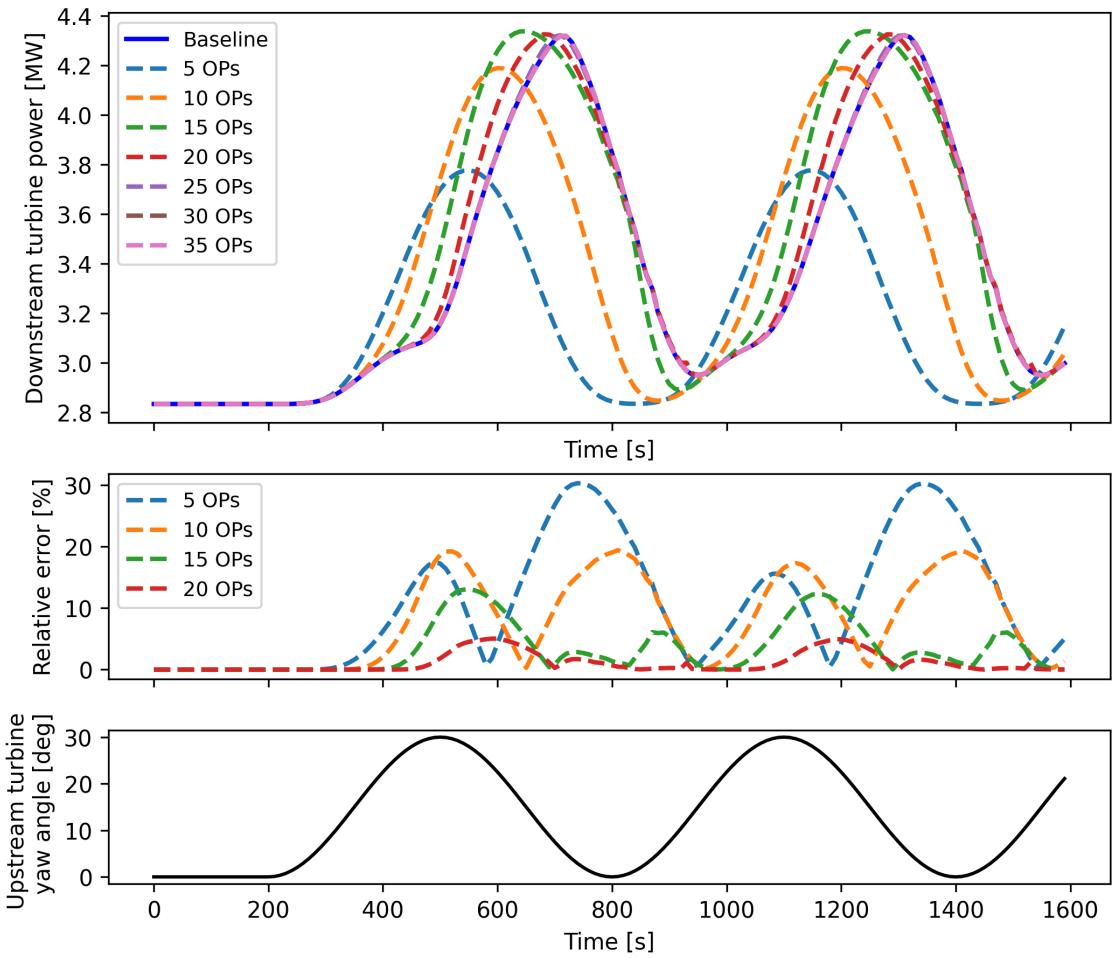
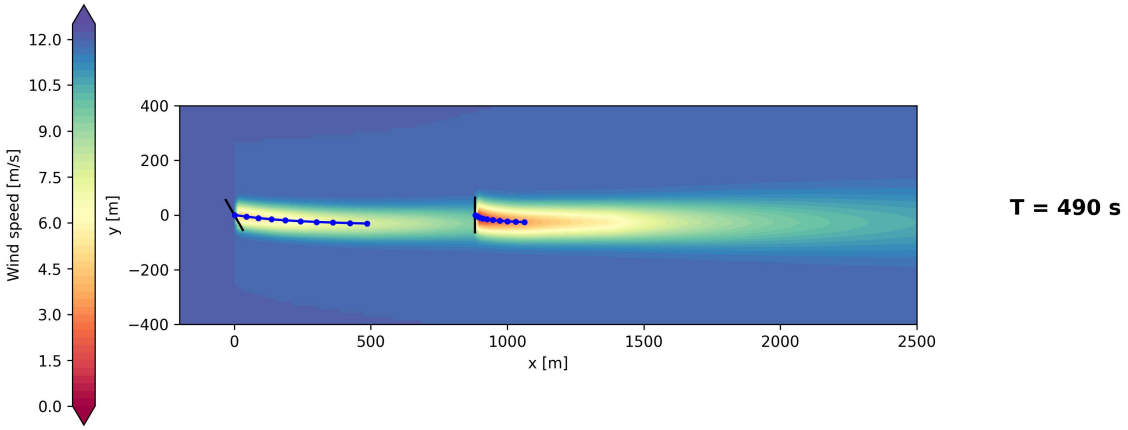


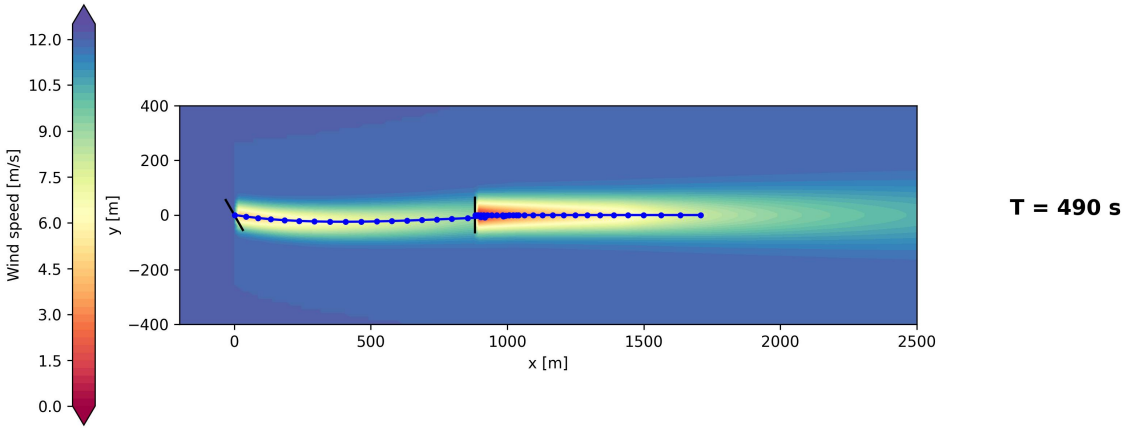
Figure 4.15: *Top:* downstream turbine power comparison obtained with different fixed OPs numbers and using the default strategy as baseline. *Bottom*: Relative error with the baseline results.

greater number of OPs than 25 ensures a perfect match with the baseline simulation results, suggesting that this number of OPs is sufficient to capture the important dynamic effects of the upstream turbine wake on the downstream turbine. Whereas, discrepancies are observed in the remaining simulations. From the analysis of the relative error trend, a decrease in accuracy is observed with a reduction in the number of OPs, with the simulation based on 5 OPs showing a significant phase shift in the maximum power output and predicting reduced benefits from the wake misalignment of the upstream turbine. It is worth noting that the simulation with 20 OPs consistently exhibits a relative error of less than 10%.

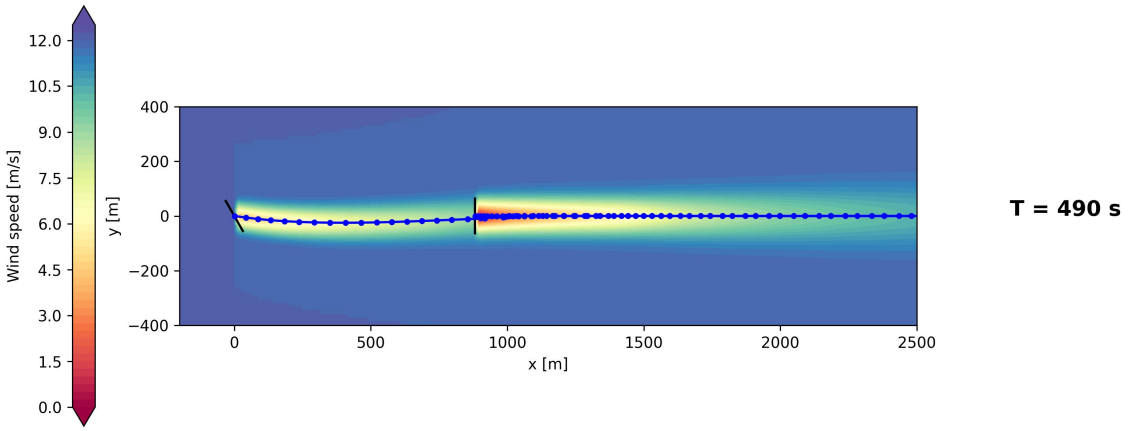
The reason for the high percentage error, especially in simulations with fewer than 15 OPs, is due to the solver's inability to capture the wake advection delay with an insufficient number of OPs. Figure 4.16, which shows the wind field at $t = 490$ s, helps to better understand this: in the simulation with 10 OPs (Figure 4.16.a), the observation point at the end of the chain fails to reach the rotor of the downstream turbine due to the reduced velocities downstream of the upstream turbine. This results in the unsteady nature of the wake being modelled only until



(a) 10 OPs



(b) 25 OPs



(c) Baseline

Figure 4.16: Contours of the wind speed at the turbine hub height at $t = 490$ s obtained with the different OPs deletion strategies: fixed OPs number versus OPs deleted outside of the domain.

OPs number	Saved time (%)
5	8.4%
10	7.9%
15	4.9%
20	4.8%
25	1.6%
30	1.5%
35	0.07%

Table 4.5: Saved computation time obtained with the proposed OPs deletion strategy.

the end of the chain; thereafter, a steady approach is adopted. In other words, after the most downstream OP, the wake deflection is computed according to the steady state formulation. This explains why the downstream turbine power output calculated with a reduced number of OPs has a less pronounced phase shift with the upstream turbine yaw angle trend. Thus, as the number of OPs decreases, the solver tends to approach a steady-state solution. On the other hand, a number of OPs equal to, for example, 25, is sufficient to characterize the upstream turbine wake over a distance that allows for the correct calculation of the downstream turbine power, as seen in the good agreement of the dynamic wake modeling between Figure 4.16.b and 4.16.c.

We report in Table 4.5 the percentual saved computation time achieved against the default OPs deletion strategy, which took 80 s to simulate 1600 s (26 min and 40 s). It turns out that as the number of OPs decreases, so does the computation time, indicating that the interpolation phase of the thrust coefficient and yaw angle values contributes to determining the computational cost in UFLORIS. With the strategy implemented here, it is possible to save up to 4.8% of the computation time while still ensuring an accuracy of over 90%. While the saved time for this simple test case with two turbines may seem small given the computational cost of the baseline simulation, for simulations of longer-duration events in large wind farms, the reduction in computation cost is significant. Additionally, the impact of the number of OPs increases significantly as the number of turbines (and thus the number of OPs chains) involved in the simulation increases. A detailed analysis of the CPU time dependence on the number of turbines would therefore be necessary to address this; however, proof of this is provided in the results of the next section, where UFLORIS is used to simulate a large number of offshore wind turbines.

In conclusion, the novel OPs deletion strategy allows for a reduction in CPU time while maintaining good accuracy. For this simple test case, the crucial factor is ensuring good modeling of the upstream wake for turbine performance calculations. Clearly, the number of OPs should be selected consistently with the specific case study. A possible approach is to determine the number of OPs such that the OPs chain is described over a distance equal to, for example, 15 diameters, in line with the assumption used in FLORIS that upstream turbines add turbulence to the ambient turbulence when positioned within a distance less than 15 diameters. This

is because in the GCH model wake recovery is related to turbulence intensity, as indicated in Equation 2.10. Similarly, it can be assumed that a distance equal to 15 diameters may ensure accuracy also with the other wake models available. Further tests, with progressively larger layouts and different inflow and turbine control conditions, should be conducted to understand if this approach yields good results. Indeed, a sensitivity study would be necessary to understand the number of downstream wind turbines strongly affected by the wake, which would require a greater number of OPs than those used for this simple test with only two turbines.

4.3 Low pressure event

Here we present the results of the simulation of the low-pressure system considering the full Belgian-Dutch offshore cluster, as described in Section 3.3. The goal is to compare the models within the UFLORIS framework to highlight the differences between them, understand how the unsteady solvers capture wake dynamics, and compare the CPU time of the models for a simulation with such a large number of turbines. Another objective is to compare the results obtained in UFLORIS with real data, to see if there is a good agreement between them and to highlight possible reasons for the mismatch in the results. The section is organized as follows: first, we compare the power output of one of the Belgian offshore wind farms calculated with the different solvers; then we compare the results obtained in UFLORIS with SCADA data.

4.3.1 Models comparison

In Figure 4.17, it is shown the comparison of the binned power outcome for the offshore wind farm computed with the models within the UFLORIS framework using a wind direction bin width of 10 degrees. In Table 4.6, it's reported the percent error of the binned farm power computed with the unsteady models, using the quasi-steady results as baseline . The error is defined as:

$$\text{Percent error} = \frac{|P_u - P_{qs}|}{P_{qs}} \cdot 100 \tag{4.2}$$

where P_{qs} is the mean power computed with QS FLORIS and P_u refers to the mean power obtained with one of the unsteady models. We use the steady model as a baseline because of the strong assumptions done in the parameter selection which, combined with unknown real operations of the turbines, makes hard to directly compare the simulation results against SCADA data. The values reported refer to results obtained before the third hour and half of the event. Indeed, after this time the models converge since the wind turbines are above rated.

The unsteady models show a good agreement not only between them but also with QS FLORIS, in which the dynamic features of the wake are neglected. From one point of view, this is valuable as proof of the fact that averaging the power of the unsteady results is coherent with a steady approach. On the other hand, the slight

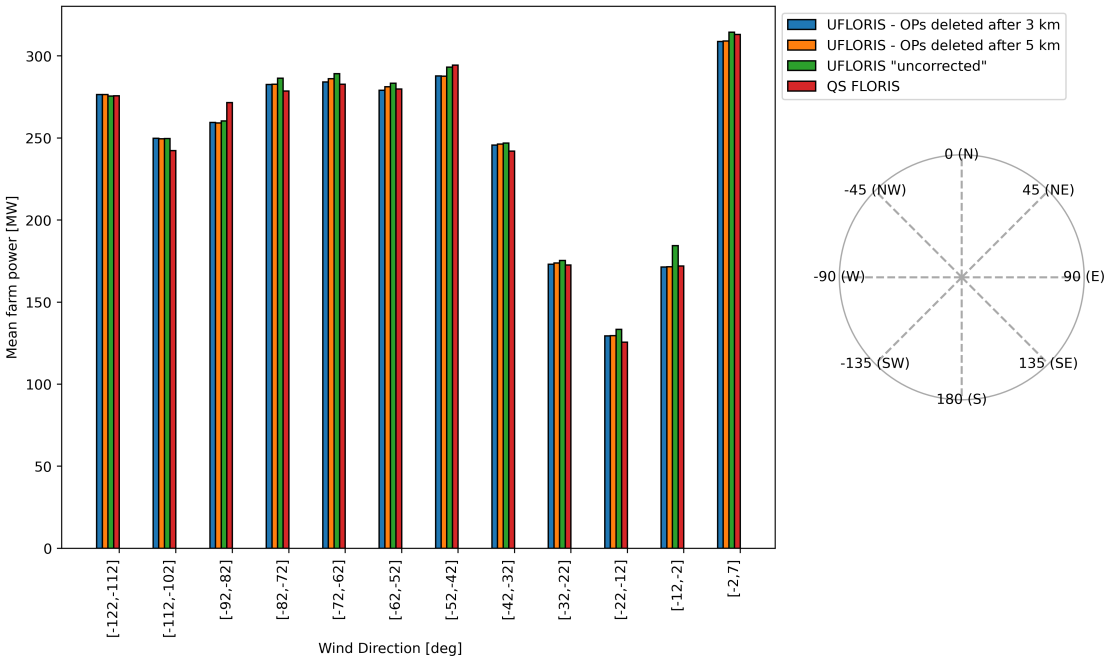


Figure 4.17: Binned farm power computed with the models within the UFLORIS framework using a wind direction bin width of 10 degrees.

Percent error	Wind direction [deg]											
	[-122,-112]	[-112,-102]	[-92,-82]	[-82,-72]	[-72,-62]	[-62,-52]	[-52,-42]	[-42,-32]	[-32,-22]	[-22,-12]	[-12,-2]	[-2,7]
UFLORIS "uncorrected"	0.07	2.99	4.14	2.78	2.31	1.25	0.41	2.03	1.56	6.23	7.23	0.46
OPs deleted after 3 km	0.24	3.10	4.44	1.41	0.51	0.31	2.23	1.53	0.20	3.10	0.33	1.38
OPs deleted after 5 km	0.24	2.99	4.54	1.49	1.21	0.48	2.26	1.76	0.65	3.24	0.22	1.30

Table 4.6: Percent error using quasi-steady results as baseline.

discrepancies between UFLORIS and QS FLORIS during the strongly dynamic condition suggest that the unsteady wake is not fully captured. A possible reason can be found in the low time resolution used as input in the UFLORIS framework. Indeed, the real data concerning the wind direction and amplitude (see Figure 3.11) are few and so probably insufficient to properly set up an unsteady simulation. One possible way to overcome this could be repeating the simulations decreasing the interpolation time step in the inflow data. Unfortunately, due to lack of time, no other simulation, with refined inflow input, has been done.

From the analysis of the results, it emerges that the unsteady solvers generally predict a higher power output, with UFLORIS "uncorrected" exhibiting a tendency to deviate more from steady results. The reasons for this trend are not clear. One possible explanation is that for the considered inflow conditions, the wakes of the upstream turbines tend to have a greater impact on the downstream turbines in the quasi-steady approach, where the wake is instantly adapted to the turbine settings and inflow conditions. The delay due to wake advection would instead result in downstream turbines seeing a cleaner flow on average. The higher differences between the "uncorrected" version and the quasi-steady solver may be attributed to the improper unsteady approach on which it is based: the model wrongly captures

Wake model	CPU time for one time step	Simulation CPU time
QS FLORIS	4.25 s	2 hours and 3 minutes
UFLORIS "uncorrected"	758.5 s (12.6 min)	15 days, 6 hours and 36 minutes
OPs deleted after 3 km	1006.3 s (16.8 min)	20 days, 6 hours and 22 minutes
OPs deleted after 5 km	1569 s (26.15 min)	31 days, 14 hours and 21 minutes

Table 4.7: CPU time for the different wake models.

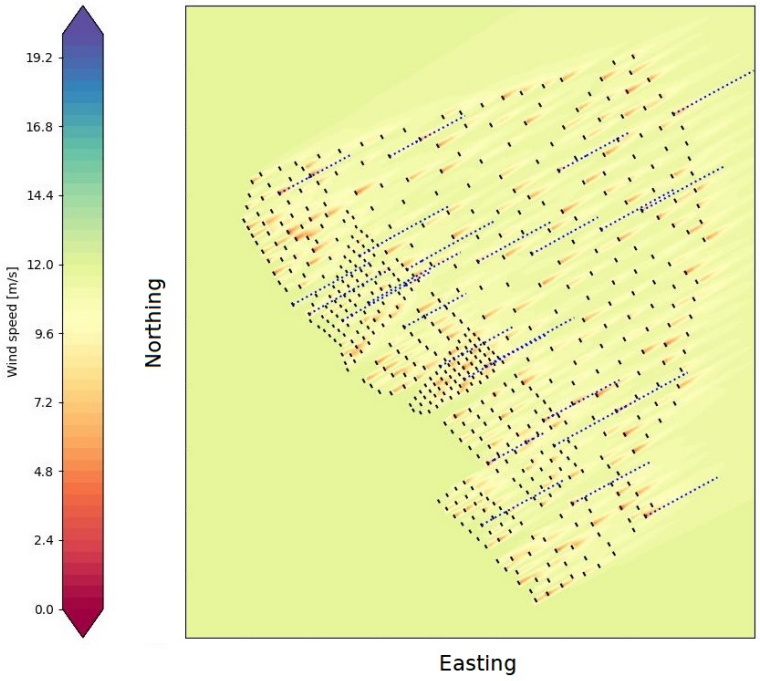
the unsteady features of the wake because at each time step the current thrust coefficient values of the turbines are used to compute the wake velocity deficit.

It is of considerable interest to compare the computation time for the different models. In Table 4.7, we report the average CPU time for resolving one time step and the total simulation CPU time requested by the simulations done in the UFLORIS framework. There is a notable discrepancy between the quasi-steady and the unsteady solvers: QS FLORIS takes few seconds to complete one loop, whereas in the UFLORIS models, both "uncorrected" and "corrected", a time of the order of tens of minutes is necessary for the resolution of each time step.

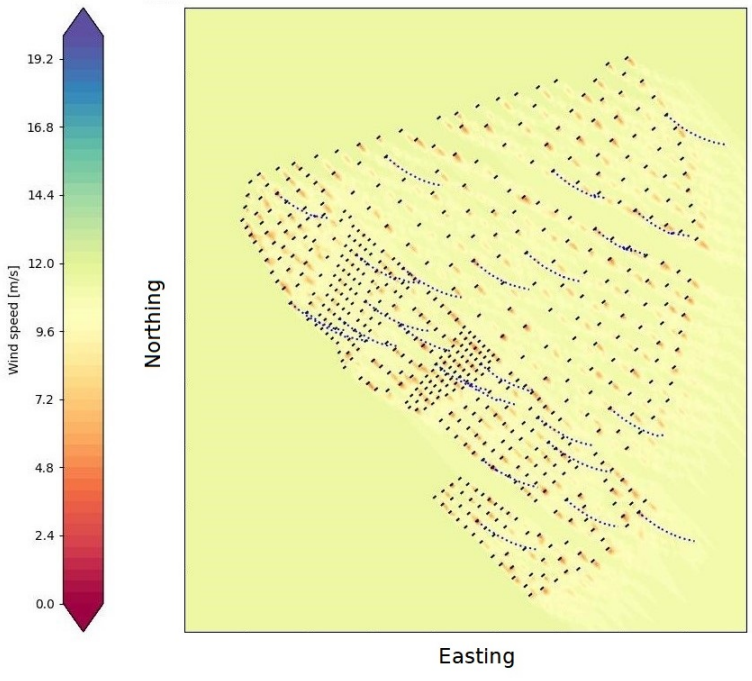
The reason for the large difference in the CPU time between the quasi-steady and the unsteady solvers can be found in computationally expensive OPs advection through the domain to account for wake advection. Indeed, in the UFLORIS models, both "corrected" and "uncorrected", the phase of convection of the OPs, as described in Equation 2.53, and the computation of the velocity at OPs locations request a loop over the turbines at each time step. Due to the large number of wind turbines the disparities in computation time between QS and UFLORIS are notable. It's also remarkable that the computational cost of the UFLORIS "corrected" simulations is greatly larger than the CPU time requested by the "uncorrected" version. The main difference between the two unsteady models is that the "corrected" version accounts for the variation of the turbine yaw angle and the thrust coefficient along the wake: OPs transport with them the turbine settings evaluated at the time they were created at the rotor and the local velocities are computed through interpolations of closest turbine setting values. This confirms a statement made in the previous test case: one of the main sources of slowing down the code is the addition of another dimension, the time, to the wake velocity deficit computation. Another proof of this is the saving time that can be achieved by implementing the OPs deletion strategy used here. Indeed, the data in table shows that reducing the distance after which the OPs are eliminated from 3 km to 5 km, allows to reduce the computational cost by up to 36%, yet retaining the same accuracy.

4.3.2 Comparison with SCADA data

Since no differences can be discerned in the results obtained with UFLORIS "corrected", in the following only the simulation with OPs deleted after 3 km is considered. Figure 4.19 shows the comparison of the farm power outcome obtained in the UFLORIS framework with SCADA data. In the figure, it is also shown the farm power neglecting wake effects, which is simply obtained by entering the turbine

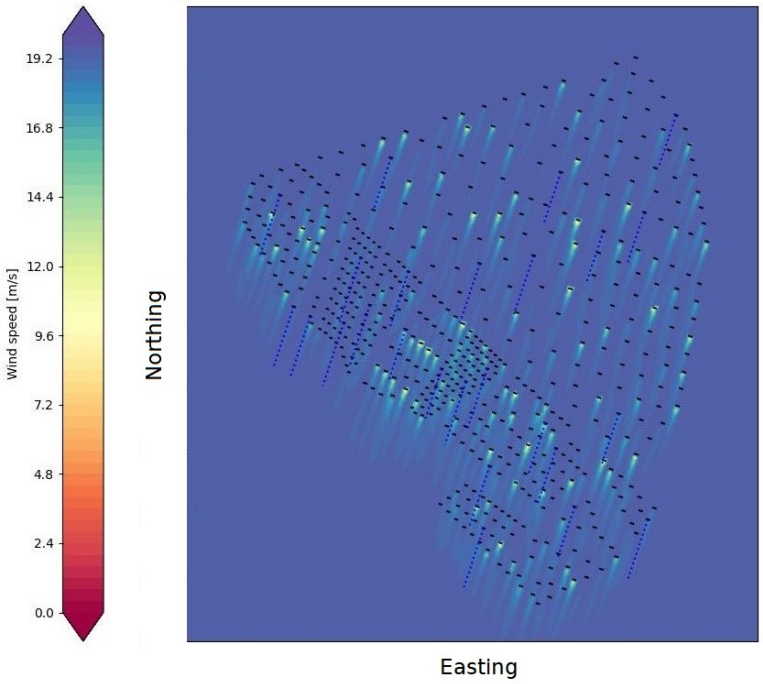


(a) Time step 340 (time 01:00).



(b) Time step 780 (time 02:50).

Figure 4.18: Contours of the instantaneous wind speed at the turbines mean hub height obtained at different time steps. The wind field is computed in the UFLORIS framework with the OPs deleted after 3 km. For the sake of clarity few OPs chains are represented in flow field.



(c) Time step 1175 (time 4:29).

Figure 4.18: Contours of the instantaneous wind speed at the turbines mean hub height obtained at different time steps. The wind field is computed in the UFLORIS framework with the OPs deleted after 3 km. For the sake of clarity few OPs chains are represented in flow field.

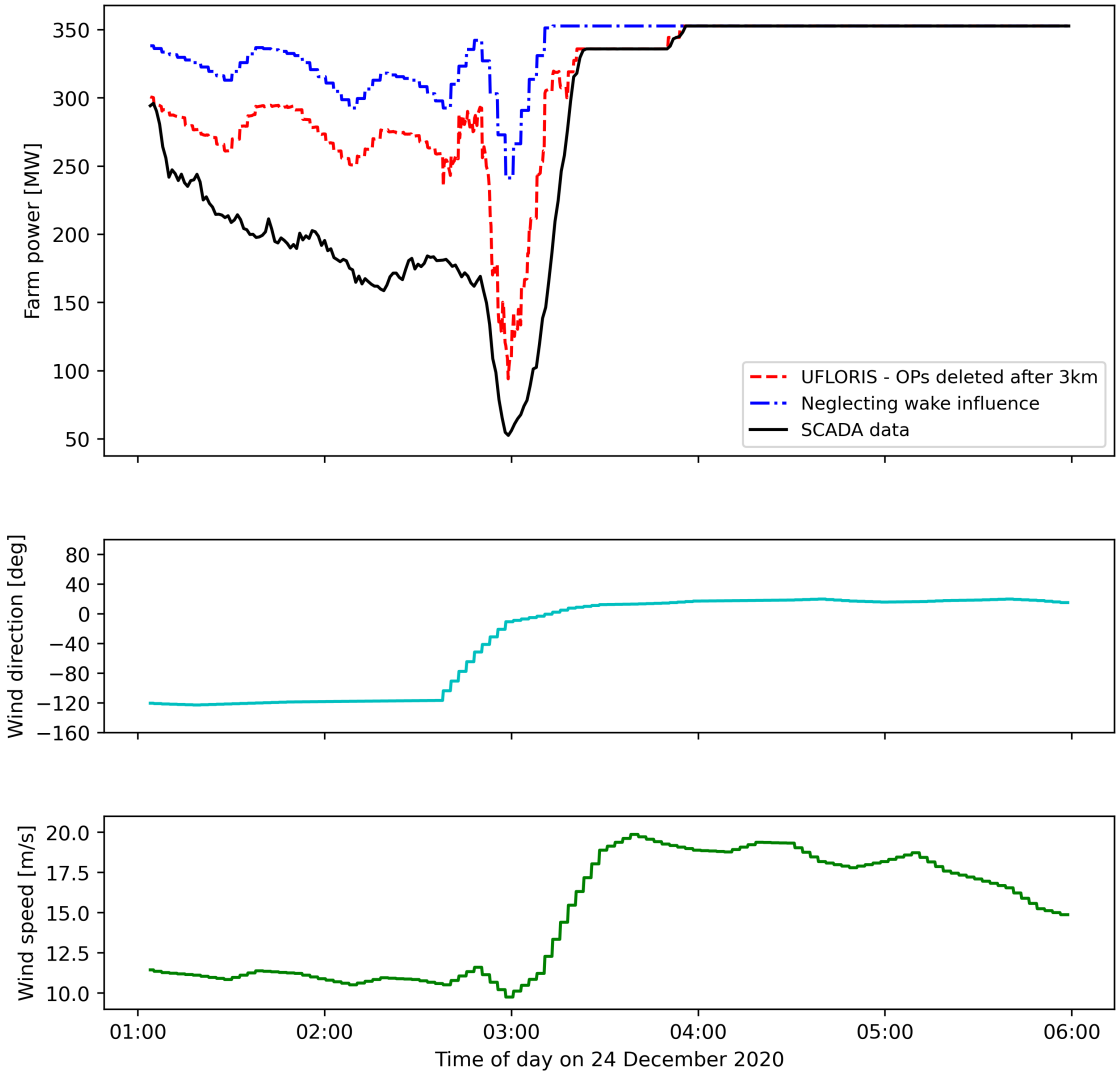


Figure 4.19: Comparison of the farm power outcome with SCADA data.

power table with the wind speed data; thus, it's also offered a steady-state solution to have an "ideal" baseline of this comparison.

The UFLORIS results show good agreement with time-series of SCADA in the strongly dynamic condition, around the third hour of the event when there are rapid changes in wind direction and speed. Here, the discrepancy may be linked to the inappropriate parameter values used, particularly the turbulence parameter set at 6%, which is evidently too low for this specific case study. Fine-tuning of the parameters, tailored to the current event, would be necessary to achieve results closer to actual data. After the fourth hour of the EWE, when the wind blows from the north, the results converge since the wind amplitude is greater than the rated speed. It is interesting to note that the turbines produce the maximum design power despite the wake influence of the neighbouring farms due to the high wind amplitude.

Instead, in the first part of the simulation, before the fast shifts in wind direction, the results obtained with UFLORIS show notable discrepancies from the real

data. These great differences in the first part of the simulation could be related, as mentioned above, to the improper selection of the wake model parameters. In addition, since the real control strategies actuated are not fully known, different turbine settings between the real data and the simulation could also contribute to increase the computation error. A significant factor negatively impacting the simulation results is the slight mismatches between the wind inflow magnitude and the actual free stream velocity. Figure 4.20 illustrates the comparison between the background flow in UFLORIS, derived from SCADA data, and the actual wind speed at the rotor of some of the south-western turbines, which serve as the upstream turbines during the time period between 01:00 and 03:00. It is evident that the inflow wind speed tends to be higher than the actual free stream velocity, leading to an overestimation of the power generated by the upstream turbines. Given that turbine power is proportional to the cube of wind speed, even slight fluctuations in wind magnitude can result in significant power discrepancies. This demonstrates one of the main drawbacks resulting from adopting a homogeneous background flow that could adequately represent the inflow conditions for all the Belgian offshore wind farms.

To sum up, although the results show an excellent trend capture in the power drop phase of the event, given the small differences in the results obtained with the various models and the discrepancies with the actual data (in particular in the first part of the EWE), it is clear that considerable progress still needs to be made. The results presented here are intended to provide only an initial insight into using UFLORIS for large-scale layouts. We are aware that achieving more accurate results would require tuning of the wake model parameters and the use of a background inflow with higher temporal and spatial resolution. A possible next step to better understand the differences between the models in the UFLORIS framework would be to repeat the simulation using a time-varying heterogeneous inflow extrapolated from a high-fidelity solver; for the comparison with real data, it would be necessary to use inflow data with higher temporal resolution acquired at different points in the layout. Thus, while for simple test cases with a limited number of turbines and for short periods of time UFLORIS shows considerable potential, for such a complex case study the capabilities of the solver need to be further investigated. Added to this is the large computational cost required for dynamic wake modeling due to the steep growth of the computational time as the number of turbines increases. A possible solution to overcome this limitation would be to consider parallelizing UFLORIS. Another improvement could be to allow the user to choose which turbine quantities are to be convected with the OPs in the "corrected" version of the solver. In fact, for this specific simulation where the turbines are always aligned with the wind direction, the interpolation phase of the yaw angle is unnecessary. A potential time-saving measure would therefore be achieved by selecting only the thrust coefficient as the variable to be transported in the domain.

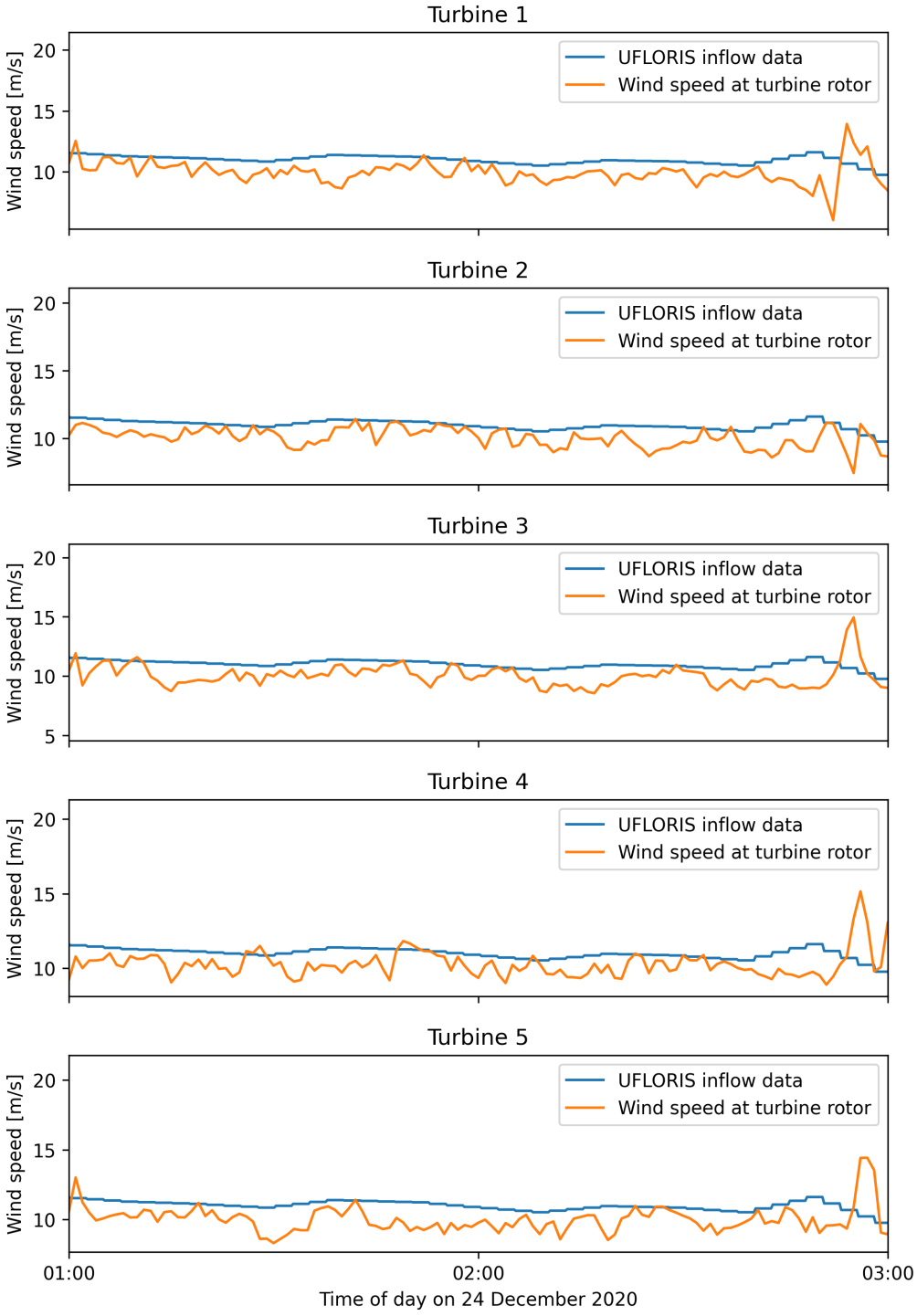


Figure 4.20: Comparison of the inflow wind used for the simulations in the UFLORIS framework (derived from SCADA data) and the real data of the wind speed at the rotor of the upstream wind turbines when the wind is blowing from south-west (from time 01:00 to 3:00 of the low pressure event).

Chapter 5

Conclusions

The goal of this thesis is to compare various wake modeling solvers, ranging from analytical engineering models with simplified physics to full-body unsteady medium-fidelity solvers; the objective is to investigate how these models represent dynamic wake behaviours while maintaining low computational costs. Additionally, this work aims to explore the capabilities of the VKI solver, UFLORIS, detailing the modifications made to improve accuracy and computational efficiency for both simplified and full offshore wind farm layouts.

The comparison between the low-fidelity unsteady solvers UFLORIS and FLORIDyn, both capable of simulating wake dynamics with low computational costs, reveals differences mainly related to the wake deficit models and the strategy used for the convection of Observation Points in the wake field. UFLORIS benefits from improvements made in the steady solver's models due to its foundation on FLORIS source code; whereas FLORIDyn lacks the ability to account for secondary steering effects and yaw-added recovery, which could enhance result fidelity in misaligned turbine scenarios. Both unsteady solvers show good agreement with higher-fidelity FAST.Farm results, and discrepancies in power output could potentially be solved through parameter tuning, a task not addressed in this study. Furthermore, both UFLORIS and FLORIDyn exhibit errors in wake advection delay due to the convection velocity of OPs in the flow field. While the VKI solver aligns closely with the higher-fidelity model, an optimal OPs convection strategy remains undetermined. Nonetheless, both engineering models are of significant interest due to their low computational costs, despite simplifications in the problem's physics.

The modifications applied to UFLORIS address issues such as incorrect calculation of the wake deficit and differences with FLORIS in the prediction of the wake centerline in steady conditions. These issues stem from ground effects in transversal wake velocities and incorrect interpolation of turbine settings convected by OPs. In addition, the novel OPs deletion strategy, based on the selection of the maximum number of OPs in the chains, reduces CPU time without significantly compromising accuracy.

Finally, simulating the low pressure event for the full Belgian-Dutch offshore cluster exposes a weakness in the UFLORIS framework. When dealing with a large number of turbines (hundreds), computational costs escalate significantly, especially in the "corrected" version of the solver. Implementing a different OPs deletion strategy helps to reduce simulation CPU time, although not enough to make this

engineering model viable for such applications. Despite computational costs and discrepancies between UFLORIS models and real data when turbine operational states are unknown, the overall power trend captures rapid wind condition changes effectively.

Bibliography

[1] GWEC – Global Wind Energy Council. *Global Wind Report 2023*. Tech. rep. 2023. URL: https://gwec.net/global-wind-report-2021/.

[2] International Energy Agency (IEA). *World Energy Outlook 2023*. Licence: CC BY 4.0 (report); CC BY NC SA 4.0 (Annex A). Paris, 2023. URL: https://www.iea.org/reports/world-energy-outlook-2023.

[3] WindEurope. *Wind energy in Europe - 2023 Statistics and the outlook for 2024-2030*. Tech. rep. Feb. 2024. URL: https://windeurope.org/intelligence-platform/product/wind-energy-in-europe-2023-statistics-and-the-outlook-for-2024-2030/.

[4] WindEurope. *Our energy, our future - How offshore wind will help Europe go carbon-neutral*. Tech. rep. Nov. 2019. URL: https://windeurope.org/intelligence-platform/product/our-energy-our-future/.

[5] P. Veers et al. "Grand challenges in the design, manufacture, and operation of future wind turbine systems". In: *Wind Energy Science* 8.7 (2023), pp. 1071–1131. DOI: 10.5194/wes-8-1071-2023.

[6] Fernando Porté-Agel, Majid Bastankhah, and Sina Shamsoddin. "Wind-Turbine and Wind-Farm Flows: A Review". In: *Boundary-Layer Meteorology* 174 (2020), pp. 1–59. DOI: 10.1007/s10546-019-00473-0.

[7] Juan Pablo Murcia Leon. "Uncertainty quantification in wind farm flow models". English. PhD thesis. Denmark, 2017.

[8] Sjoerd Boersma et al. "A tutorial on control-oriented modeling and control of wind farms". In: May 2017, pp. 1–18. DOI: 10.23919/ACC.2017.7962923.

[9] *FLORIS v3.4.1*. Accessed March 30, 2024. URL: https://github.com/NREL/floris.

[10] Pedersen, Mads M. and Forsting, Alexander Meyer and van der Laan, Paul and Riva, Riccardo and Alcayaga Roman, Leonardo A. and Criado Risco, Javier and Friis-Møller, Mikkel and Quick, Julian and Christiansen, Jens Peter Schøler and Valotta Rodrigues, Rafael and Olsen, Bjarke Tobias and Réthoré, Pierre-Elouan. "PyWake 2.5.0: An open-source wind farm simulation tool". In: (Feb. 2023). URL: https://gitlab.windenergy.dtu.dk/TOPFARM/PyWake.

[11] M. Becker et al. "The revised FLORIDyn model: implementation of heterogeneous flow and the Gaussian wake". In: *Wind Energy Science* 7.6 (2022), pp. 2163–2179. DOI: 10.5194/wes-7-2163-2022.

[12] Benoit Foloppe et al. "Development of a dynamic wake model accounting for wake advection delays and mesoscale wind transients". In: *Journal of Physics: Conference Series* 2265 (May 2022), p. 022055. DOI: 10.1088/1742-6596/2265/2/022055.

[13] E. et al Bossanyi. *D1.2: Description of the reference and the control-oriented wind farm models*. CL-Windcon. 2018. URL: https://www.clwindcon.eu/public-deliverables/.

[14] Gunner Chr. Larsen et al. *Dynamic wake meandering modeling*. English. Denmark. Forskningscenter Risoe. Risoe-R 1607(EN). Risø National Laboratory, 2007. ISBN: 978-87-550-3602-4.

[15] Sjoerd Boersma et al. "A control-oriented dynamic wind farm flow model: "WFSim"". In: *Journal of Physics: Conference Series* 753 (Oct. 2016). DOI: 10.1088/1742-6596/753/3/032005.

[16] National Renewable Energy Laboratory. *SOWFA*. Accessed March 30, 2024. URL: https://www.nrel.gov/wind/nwtc/sowfa.html.

[17] Sandia National Laboratories Lawrence Berkeley National Laboratory National Renewable Energy Laboratory. *AMR-Wind*. Accessed March 30, 2024. URL: https://github.com/Exawind/amr-wind.

[18] Paul Bartholomew et al. "Xcompact3D: An open-source framework for solving turbulence problems on a Cartesian mesh". In: *SoftwareX* 12 (June 2020), p. 100550. DOI: 10.1016/j.softx.2020.100550.

[19] Filippo Campagnolo et al. "Further calibration and validation of FLORIS with wind tunnel data". In: *Journal of Physics: Conference Series* 2265 (May 2022), p. 022019. DOI: 10.1088/1742-6596/2265/2/022019.

[20] Pieter Gebraad et al. "Wind plant power optimization through yaw control using a parametric model for wake effects—A CFD simulation study". In: *Wind Energy* (Dec. 2014). DOI: 10.1002/we.1822.

[21] Bart Doekemeijer, J. W. Wingerden, and Paul Fleming. "A tutorial on the synthesis and validation of a closed-loop wind farm controller using a steady-state surrogate model". In: July 2019, pp. 2825–2836. DOI: 10.23919/ACC.2019.8815126.

[22] Nicholas Baker et al. "Best Practices for Wake Model and Optimization Algorithm Selection in Wind Farm Layout Optimization". In: Jan. 2019. DOI: 10.2514/6.2019-0540.

[23] Majid Bastankhah and Fernando Porté-Agel. "Experimental and theoretical study of wind turbine wakes in yawed conditions". In: *Journal of Fluid Mechanics* 806 (Nov. 2016), pp. 506–541. DOI: 10.1017/jfm.2016.595.

[24] Jason Jonkman and Kelsey Shaler. *FAST.Farm User's Guide and Theory Manual*. Tech. rep. NREL/TP-5000-78785. Golden, CO: National Renewable Energy Laboratory, 2021. URL: https://www.nrel.gov/docs/fy21osti/78485.pdf.

[25] Pieter Gebraad et al. "A Data-Driven Model for Wind Plant Power Optimization by Yaw Control". In: June 2014. DOI: 10.1109/ACC.2014.6859118.

[26] Alfredo Peña, Pierre-Elouan Réthoré, and M. Paul van der Laan. "On the application of the Jensen wake model using a turbulence-dependent wake decay coefficient: The Sexbierum case". In: *Wind Energy* 19 (Mar. 2015). DOI: 10.1002/we.1863.

[27] N.O. Jensen. *A note on wind generator interaction.* English. Risø-M 2411. Risø National Laboratory, 1983. ISBN: 87-550-0971-9.

[28] Angel Jimenez, Antonio Crespo, and Emilio Migoya. "Application of a LES technique to characterize the wake deflection of a wind turbine in yaw". In: *Wind Energy* 13 (Sept. 2009), pp. 559–572. DOI: 10.1002/we.380.

[29] Majid Bastankhah and Fernando Porté-Agel. "A new analytical model for wind-turbine wakes". In: *Renewable Energy* 70 (2014). Special issue on aerodynamics of offshore wind energy systems and wakes, pp. 116–123. ISSN: 0960-1481. DOI: https://doi.org/10.1016/j.renene.2014.01.002.

[30] Amin Niayifar and Fernando Porté-Agel. "A new analytical model for wind farm power prediction". In: *Journal of Physics: Conference Series* 625.1 (June 2015), p. 012039. DOI: 10.1088/1742-6596/625/1/012039.

[31] L. A. Martínez-Tossas et al. "The aerodynamics of the curled wake: a simplified model in view of flow control". In: *Wind Energy Science* 4.1 (2019), pp. 127–138. DOI: 10.5194/wes-4-127-2019.

[32] Antonio Crespo and Julio Hernández. "Turbulence characteristics in wind-turbine wakes". In: *Journal of Wind Engineering and Industrial Aerodynamics* 61 (June 1996), pp. 71–85. DOI: 10.1016/0167-6105(95)00033-X.

[33] J. King et al. "Control-oriented model for secondary effects of wake steering". In: *Wind Energy Science* 6.3 (2021), pp. 701–714. DOI: 10.5194/wes-6-701-2021.

[34] Christopher J. Bay et al. "Addressing deep array effects and impacts to wake steering with the cumulative-curl wake model". In: *Wind Energy Science* (2023).

[35] Majid Bastankhah et al. "Analytical solution for the cumulative wake of wind turbines in wind farms". In: *Journal of Fluid Mechanics* (Oct. 2020). DOI: 10.1017/jfm.2020.1037.

[36] F. Blondel and M. Cathelain. "An alternative form of the super-Gaussian wind turbine wake model". In: *Wind Energy Science* 5.3 (2020), pp. 1225–1236. DOI: 10.5194/wes-5-1225-2020.

[37] I. Katic, J. Højstrup, and N.O. Jensen. "A Simple Model for Cluster Efficiency". English. In: *EWEC'86. Proceedings. Vol. 1.* Ed. by W. Palz and E. Sesto. European Wind Energy Association Conference and Exhibition, EWEC '86 ; Conference date: 06-10-1986 Through 08-10-1986. A. Raguzzi, 1987, pp. 407–410.

[38] Pieter Gebraad and J. W. Wingerden. "A Control-Oriented Dynamic Model for Wakes in Wind Plants". In: *Journal of Physics: Conference Series* 524 (June 2014), p. 012186. DOI: 10.1088/1742-6596/524/1/012186.

[39] Marcus Becker. *Gaussian FLORIDyn, Matlab implementation belonging to the paper: The revised FLORIDyn model: Implementation of heterogeneous flow and the Gaussian wake*. URL: https://data.4tu.nl/articles/software/Gaussian_FLORIDyn_Matlab_implementation_belonging_to_the_paper_The_revised_FLORIDyn_model_Implementation_of_heterogeneous_flow_and_the_Gaussian_wake/19867846/1 (last access: 24 November 2023). 2022. DOI: 10.4121/19867846.v1.

[40] I. Katic, J. Højstrup, and N.O. Jensen. "A Simple Model for Cluster Efficiency". English. In: *EWEC'86. Proceedings. Vol. 1*. Ed. by W. Palz and E. Sesto. European Wind Energy Association Conference and Exhibition, EWEC '86 ; Conference date: 06-10-1986 Through 08-10-1986. A. Raguzzi, 1987, pp. 407–410.

[41] A. Farrell et al. "Design and analysis of a wake model for spatially heterogeneous flow". In: *Wind Energy Science* 6.3 (2021), pp. 737–758. DOI: 10.5194/wes-6-737-2021.

[42] S. A. Hsu, Eric A. Meindl, and David B. Gilhousen. "Determining the Power-Law Wind-Profile Exponent under Near-Neutral Stability Conditions at Sea". In: *Journal of Applied Meteorology and Climatology* 33.6 (1994), pp. 757–765. DOI: https://doi.org/10.1175/1520-0450(1994)033<0757:DTPLWP>2.0.CO;2.

[43] *Offshore vertical wind shear: Final report on NORSEWInD's work task 3.1*. English. DTU Wind Energy E 0005. Denmark: DTU Wind Energy, 2012.

[44] Helmut Vogel. "A better way to construct the sunflower head". In: *Mathematical Biosciences* 44.3 (1979), pp. 179–189. ISSN: 0025-5564. DOI: https://doi.org/10.1016/0025-5564(79)90080-4.

[45] G. I. Taylor. "The Spectrum of Turbulence". In: *Proceedings of the Royal Society of London. Series A, Mathematical and Physical Sciences* 164.919 (1938), pp. 476–490. ISSN: 00804630.

[46] *OpenFAST v3.5.2*. Accessed March 30, 2024. URL: https://github.com/OpenFAST/openfast.

[47] NREL. *OpenFAST Documentation*. Accessed March 30, 2024. URL: https://openfast.readthedocs.io/en/dev/index.html.

[48] Gunner Larsen et al. "Wake meandering: A pragmatic approach". In: *Wind Energy* 11 (July 2008), pp. 377–395. DOI: 10.1002/we.267.

[49] J.F. Ainslie. "Calculating the flowfield in the wake of wind turbines". In: *Journal of Wind Engineering and Industrial Aerodynamics* 27.1 (1988), pp. 213–224. ISSN: 0167-6105. DOI: https://doi.org/10.1016/0167-6105(88)90037-2.

[50] B. J. Jonkman. *TurbSim User's Guide v2.00.00*. Tech. rep. Golden, CO: National Renewable Energy Laboratory, 2014. URL: https://openfast.readthedocs.io/en/dev/source/user/turbsim/index.html.

[51] G. I. Taylor. "The Spectrum of Turbulence". In: *Proceedings of the Royal Society of London Series A* 164.919 (Feb. 1938), pp. 476–490. DOI: 10.1098/rspa.1938.0032.

[52] John Jasa et al. "Effectively using multifidelity optimization for wind turbine design". In: *Wind Energy Science* 7 (May 2022), pp. 991–1006. DOI: 10.5194/wes-7-991-2022.

[53] Jennifer Annoni et al. "Analysis of axial-induction-based wind plant control using an engineering and a high-order wind plant model". In: *Wind Energy* 19 (Aug. 2015). DOI: 10.1002/we.1891.

[54] Jonkman JM et al. *Definition of a 5MW Reference Wind Turbine for Offshore System Development*. Tech. rep. National Renewable Energy Laboratory (NREL), 2009.

[55] *OpenFAST repository: r-test*. Accessed March 30, 2024. URL: https://github.com/OpenFAST/r-test.

[56] Paula Doubrawa, Jennifer Annoni, and Jason Jonkman. "Optimization-Based Calibration of FAST.Farm Parameters against Large-Eddy Simulations". In: Jan. 2018. DOI: 10.2514/6.2018-0512.

[57] International Electrotechnical Commission. *IEC 61400-1 Wind turbine generator systems-Part 1: Safety requirements*. Tech. rep. Geneva, Switzerland: International Electrotechnical Commission, 1999.

[58] International Electrotechnical Commission. *IEC 61400-1 Wind turbines-Part 1: Design requirements*. Tech. rep. Geneva, Switzerland: International Electrotechnical Commission, Aug. 2005.

[59] A. Vemuri et al. "Sensitivity analysis of mesoscale simulations to physics parameterizations over the Belgian North Sea using Weather Research and Forecasting – Advanced Research WRF (WRF-ARW)". In: *Wind Energy Science* 7.5 (2022), pp. 1869–1888. DOI: 10.5194/wes-7-1869-2022. URL: https://wes.copernicus.org/articles/7/1869/2022/.

[60] *Climate & Meteorology - GOV.UK*. https://assets.publishing.service.gov.uk/media/623332cd8fa8f504a584cfd1/Appendix_1f_-_Climate___Meteorology.pdf. Accessed: 13 March 2024.

[61] *Belgian offshore platform*. https://www.belgianoffshoreplatform.be/en/. Accessed: 30/03/2024.

[62] *Borssele wind farm zone*. https://www.noordzeeloket.nl/en/functions-and-use/offshore-wind-energy/free-passage-shared-use/borssele-wind-farm-zone/. Accessed: 30/03/2024.

[63] *Borssele I&II*. https://english.rvo.nl/subsidies-financing/offshore-wind-energy/borssele-sites-i-ii. Accessed: 30/03/2024.

[64] *Borssele III&IV*. https://english.rvo.nl/subsidies-financing/offshore-wind-energy/borssele-sites-iii-iv. Accessed: 30/03/2024.

[65] *Borssele V*. https://english.rvo.nl/subsidies-financing/offshore-wind-energy/borssele-site-v. Accessed: 30/03/2024.

[66] L. A. Martínez-Tossas et al. "The aerodynamics of the curled wake: a simplified model in view of flow control". In: *Wind Energy Science* 4.1 (2019), pp. 127–138. DOI: 10.5194/wes-4-127-2019.

[67] Carl Shapiro, Dennice Gayme, and Charles Meneveau. "Modelling yawed wind turbine wakes: A lifting line approach". In: *Journal of Fluid Mechanics* 841 (Apr. 2018). DOI: 10.1017/jfm.2018.75.