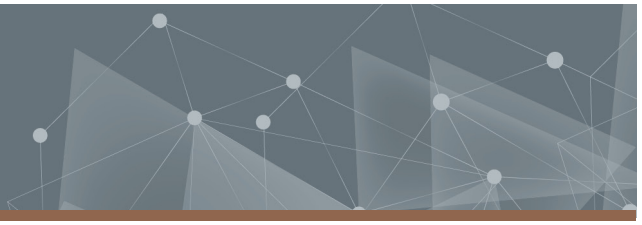
CHALMERS
UNIVERSITY OF TECHNOLOGY



Validation of a wake model for vertical-axis wind turbine farms

Master's thesis in Engineering Mathematics and Computational Science

GUSTAV VALLBO

DEPARTMENT OF MECHANICS AND MARITIME SCIENCES

CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2023
www.chalmers.se

Master's thesis 2023

Validation of a wake model for vertical-axis wind turbine farms

GUSTAV VALLBO

CHALMERS
UNIVERSITY OF TECHNOLOGY

Department of Mechanics and Maritime Sciences
Division of Fluid Dynamics
Chalmers University of Technology
Gothenburg, Sweden 2023

Validation of a wake model for vertical-axis wind turbine farms
GUSTAV VALLBO

Supervisor: Rémi Corniglion, SeaTwirl
Examiner: Håkan Nilsson, Department of Mechanics and Maritime Sciences, Division of Fluid Dynamics, Chalmers University of Technology

Master's Thesis 2023
Department of Mechanics and Maritime Sciences
Division of Fluid Dynamics
Chalmers University of Technology
SE-412 96 Gothenburg
Telephone +46 31 772 1000

Typeset in LaTeX
Gothenburg, Sweden 2023

Validation of a wake model for vertical-axis wind turbine farms
GUSTAV VALLBO
Department of Mechanics and Maritime Sciences
Chalmers University of Technology

Abstract

An analytical wake model can be used to predict the power generated by a vertical-axis wind turbine farm. The accuracy of a specific wake model is investigated in this study. In particular, the velocity field within the farm is analyzed since it determines the power that can be generated by each turbine. On a two-turbine layout, the velocity field obtained with the wake model is compared to that given by a computational fluid dynamics (CFD) simulation. The two-dimensional CFD simulation is performed with an actuator line model of the turbine blades and Reynolds-averaged Navier-Stokes turbulence modeling. One of the major findings is that the wake model results are highly dependent on the choice of combination model used to superpose wakes of multiple turbines. The results also show that one of the combination models has a better agreement with the CFD simulation. However, this comparison is limited by the accuracy of the turbulence modeling within the CFD simulation. Despite this limitation, this study is a step towards accurately modeling a vertical-axis wind farm.

Keywords: vertical-axis wind turbine (VAWT), wake model, wind farm

Acknowledgements

I want to thank my knowledgeable, energetic, and dedicated supervisor Rémi Corniglion for the support during this work. You inspired and guided me all throughout the project and I am happy that you motivated me to contribute to an open-source code. I also greatly appreciate the friendly environment at SeaTwirl. Further, I acknowledge the time and effort spent by Håkan Nilsson to develop the course *Basic usage of OpenFOAM*. Access to this material was crucial when I was getting started with OpenFOAM.

The computations were enabled by resources provided by Chalmers e-Commons.

Gustav Vallbo, Gothenburg, November 2023

Contents

1

Introduction

1.1 Background

The EU strategy on offshore wind is to increase the installed capacity in the EU to 60 GW by 2030 and to 300 GW by 2050 [1]. In May 2022, the installed capacity was about 22 GW [2]. In Sweden, there is also a great interest in offshore wind. Currently, projects amounting to 67 TWh/year are in the process of obtaining required permits [2]. This is about 40 % of Sweden's current yearly production of electricity (166 TWh as of 2021 [3]).

Vertical-axis wind turbines (VAWTs) are being developed as an alternative to the traditional horizontal-axis wind turbines (HAWTs). The wake dynamics differ between the two types. While horizontal-axis turbines require significant spacing in a wind farm to not block each other (i.e. to remain efficient), some studies show that VAWTs actually benefit from being closely spaced [4, 5]. Thus, a VAWT farm may have the potential to achieve a higher power density [W/m^2] than a HAWT farm [4].

There is a need to optimize the layout of VAWT farms. For this, different types of computational fluid dynamics (CFD) simulations can be used, as well as lower order methods such as wake models. Regarding CFD, popular choices of modeling the turbine include blade-resolved simulations and the actuator line model (ALM). In the latter, each turbine blade is *modeled* such that the boundary layer around the blade does not need to be resolved, which in turn reduces the computational cost significantly. However, the CFD simulations are still costly, making wake models an interesting alternative. The wake model of interest in this thesis was developed by Ouro & Lazennec [6] from the Reynolds-averaged Navier-Stokes (RANS) equations, assuming, among other things, inviscid flow. It gives the streamwise component of the velocity field in the wake of an isolated turbine as an analytical function. Further, such wakes can be superposed (or combined) to model a wind farm. If validated soundly, wake models could be a computationally cheap way to optimize the layout of VAWT farms to maximize power output.

1.2 Aim

The aim of the thesis is to validate the super-Gaussian wake model by Ouro & Lazennec [6] for VAWT farms (arrays consisting of multiple turbines). As a part of this, the performance of different wake combination models is evaluated.

1.3 Research questions

The focus of the thesis is to answer:

- Considering a limited number of VAWT arrays, how well does the wake model predict the wake velocity field and the power generated by each turbine in comparison with CFD (RANS+ALM) simulations?
  - How sensitive are the CFD simulations to: spatial and temporal resolution, actuator line parameters (such as the Gaussian width used to apply turbine blade forces to the flow), and inlet boundary conditions for turbulence properties?
  - How sensitive is the wake model to the choice of combination model?

1.4 Limitations

- The CFD simulations are two-dimensional, which means that the effect of the atmospheric boundary layer is neglected as well as the effect of vortices emitted from the turbine blade-tips.
- As described later in section 3.4, a single-turbine case is used to validate the CFD simulations. A sensitivity analysis is performed on this case and it is assumed that the result can be generalized to set up the remaining simulations.
- The CFD sensitivity analysis focuses on mesh-size and Gaussian width rather than time step size, since ALM simulations have shown to be insensitive to the latter parameter; see [7] for RANS and [8] for a large eddy simulation (LES).
- Only one turbine geometry and one tip-speed ratio is considered.

2

Theory

2.1 Basics of a vertical-axis wind turbine

A sketch of a two-bladed vertical-axis wind turbine is shown in figure 2.1. The whole turbine rotates about its center vertical axis. In the figure, $D = 2R$ is the turbine diameter, H is the length of the vertical turbine blades, $\boldsymbol{U}_{local}$ is the local velocity of the incoming air to the blade, $\boldsymbol{U}_{blade}$ is the tangential velocity of the blade, $\boldsymbol{U}_{rel}$ is the velocity of the air relative to the blade, $\boldsymbol{F}_l$ is the lift force, $\boldsymbol{F}_d$ is the drag force, and θ is the azimuth angle. Bold notation is used for vectors while non-bold is used for the corresponding vector-magnitudes.

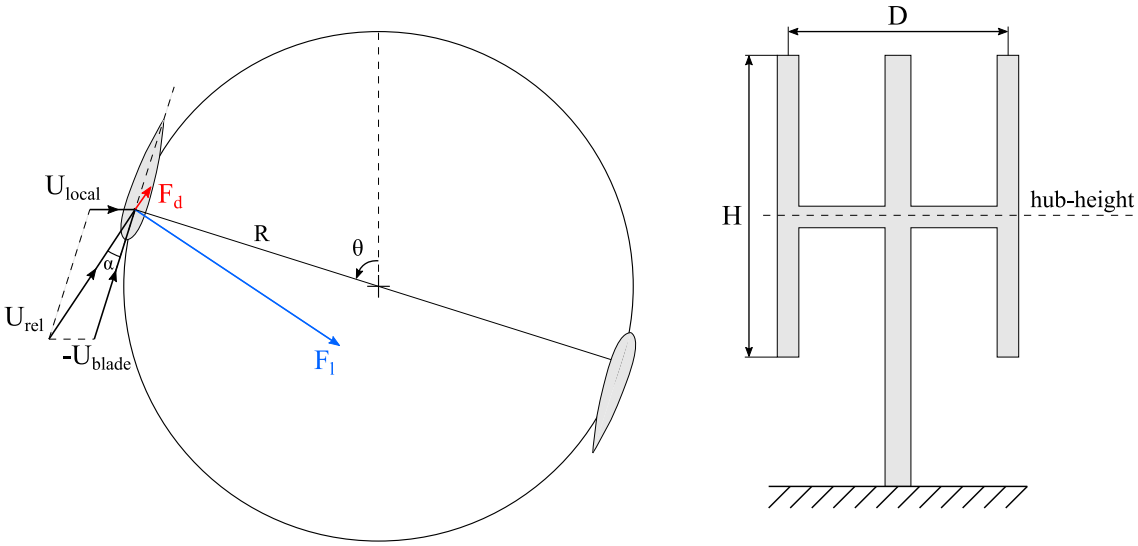


Figure 2.1: Simplified sketch of a two-bladed vertical-axis wind turbine. Right: Turbine cross-section in a vertical plane. Left: Top view of the two blades which have an airfoil profile. The wind direction is from left to right, and the airfoils and drag force are enlarged to increase visibility. The left pane is inspired by Abkar [9].

Turbine operating conditions are largely dependent on the tip-speed ratio λ, which is defined by (2.1) for a turbine operating in isolation. Importantly, the turbine thrust and power coefficient, $C_T(\lambda)$ and $C_P(\lambda)$, depend on the tip-speed ratio. Simply put, $C_T \in [0, 1]$ indicates how much the turbine obstructs the flow. A higher C_T

usually means a lower streamwise velocity in the region behind the turbine (called the *wake*). Further, the power coefficient is defined as the power generated by the turbine relative to the available power in the incoming wind.

$$\lambda = \frac{U_{blade}}{U_\infty} = \frac{\omega R}{U_\infty} \tag{2.1}$$

where U_∞ is the freestream velocity and ω is the rotation speed.

In figure 2.1, the relative velocity is given as $\boldsymbol{U}_{rel} = \boldsymbol{U}_{local} - \boldsymbol{U}_{blade}$. The angle of attack α can then be defined as the angle between the relative velocity and the chord-line of the airfoil. Note that, at the blade position shown in figure 2.1, the lift force $\boldsymbol{F}_l$ generates a torque in the counter-clockwise direction. This makes the turbine rotate and allows power to be extracted.

3

Methods

3.1 The super-Gaussian wake model

3.1.1 Model definition

The analytical super-Gaussian wake model for VAWTs is a computationally cheap alternative to CFD simulations. It gives the streamwise component of the velocity field in the wake of an isolated turbine as an analytical function. The result is usually presented as a normalized velocity deficit, defined as

$$\frac{\Delta U(x,y,z)}{U_\infty} = \frac{U_\infty - U(x,y,z)}{U_\infty}$$

where $U(x,y,z)$ is the streamwise velocity and U_∞ is the freestream velocity.

For a full derivation of the super-Gaussian model, the reader is referred to Ouro & Lazennec [6]. Here, we summarize their work by providing the mathematical formulation of the model. Let x, y, and z be coordinates in the streamwise, cross-stream, and vertical direction respectively. "At any streamwise location x behind the turbine, the velocity deficit in the yz-plane is given by multiplying the maximum deficit $C = C(x)$ with two super-Gaussian shape functions" [10] denoted $f_y(x,y)$ and $f_z(x,z)$:

$$\frac{\Delta U}{U_\infty} = C(x) f_y(x,y) f_z(x,z) = C(x) \exp\left(-\frac{|\widetilde{y}|^{n_y}}{2\widetilde{\sigma}_y^2}\right) \exp\left(-\frac{|\widetilde{z}|^{n_z}}{2\widetilde{\sigma}_z^2}\right) \tag{3.1}$$

Intuitively, the wake recovers in the streamwise direction, i.e. $C(x) \to 0$ as $x \to \infty$. This is caused by turbulent mixing between the lower-velocity air in the wake and the higher-velocity ambient air.

The terms involved in (3.1) are defined in the following. First, let (x_c, y_c, z_c) be the location of the turbine hub and note that wake model is valid for $x \geq x_c$ (i.e. behind the turbine). Normalized coordinates are written as

$$\widetilde{y} = \frac{y - y_c}{D}, \quad \widetilde{z} = \frac{z - z_c}{H}$$

where D is the turbine diameter and H is the length of the vertical turbine blades.

At a given streamwise position x, the wake extends to infinity in both the cross-stream and vertical direction. More formally, the velocity deficit in (3.1) tends to 0 as $|\widetilde{y}| \to \infty$ or $|\widetilde{z}| \to \infty$. However, the spread of the wake in y and z, hereafter called *characteristic wake widths* $\widetilde{\sigma}_y$ and $\widetilde{\sigma}_z$, can be defined according to (3.2). Note that the characteristic wake widths increase linearly in the streamwise direction.

$$\widetilde{\sigma}_y = k_y^* \frac{x - x_c}{D} + \varepsilon_y, \quad \widetilde{\sigma}_z = k_z^* \frac{x - x_c}{H} + \varepsilon_z \tag{3.2}$$

where ε_y and ε_z are initial characteristic wake widths which are closely related to the turbine cross-section (as seen by the incoming wind) $A = DH$. Further, $k_y^* = 0.50 I_u$ and $k_z^* = 0.50 I_u$ are the wake expansion rates[§] which are proportional to the streamwise turbulence intensity I_u.

Finally, the exponents n_y and n_z determine the shape of the wake. They can be expressed as

$$n_y = a_y \exp\left(-b_y \frac{x - x_c}{D}\right) + c_y, \quad n_z = a_z \exp\left(-b_z \frac{x - x_c}{H}\right) + c_z$$

where a_y, b_y, c_y, a_z, b_z and c_z are model parameters. Default values of these parameters are used in this work, as given by Ouro & Lazennec [6].

The definition of the maximum deficit $C(x)$ and the initial characteristic wake widths ε_y and ε_z are harder to interpret and are therefore offloaded to appendix A.

3.1.2 Implementation in an open-source framework

As part of this work, the super-Gaussian model is implemented in FLORIS [11] which is an open-source framework for wind turbine wake models. In FLORIS, a *wake model* actually consists of a *velocity model* (which would be the super-Gaussian model in this case), a *turbulence model*, a *deflection model*, and a *combination model*. In the wake model calculations of this work, the turbulence intensity of the wind is assumed to be constant in space, meaning that no turbulence model is used. No deflection model is used either. To model a wind farm, each turbine in the farm is assumed to operate in isolation (but the inlet conditions may be affected by upstream turbines) such that the wake of every turbine can be calculated with a velocity model like (3.1). A combination model is then used to combine the velocity model results and thus obtain the full velocity field within the farm. Two combination models are assessed in this work (defined in the next section).

This works' contribution to FLORIS include the super-Gaussian model [12], VAWT infrastructure [13], and a quick way to sample and plot velocity deficit profiles [14]. A minimal code example of the latter is available in appendix B.

The added VAWT infrastructure makes sure that the incoming velocity to a given turbine, U_{in}, is averaged over a suitable turbine cross-section area. The incoming

[§]Two different expansion rates, $k_y^* = k_z^* = 0.45 I_u$ and $k_y^* = k_z^* = 0.50 I_u$, can be found in the article by Ouro & Lazennec [6], but it has been confirmed (through private communication with P. Ouro in August 2023) that the latter was used to produce the results shown in the article.

velocity is important for multiple reasons. Firstly, in FLORIS, the turbine thrust and power coefficient, C_T and C_P, generally depend on U_{in}. Secondly, U_{in} is used to calculate the turbine power as $P = \frac{1}{2}\rho A U_{in}^3 C_P$.

In FLORIS, the simplest way of sampling U_{in} for a HAWT is with a `TurbineGrid` (see figure 3.1 left). The existing code uses a square grid with a width ratio $\alpha_{HAWT} = 0.5$ and a height ratio $\beta_{HAWT} = 0.5$. These ratios are used to ensure that the grid is within the rotor frontal area. Support for VAWTs is added to the `TurbineGrid` class by allowing a rectangular grid with different number of sample points in the cross-stream and vertical direction (making it possible to have an approximately equidistant grid for VAWTs). In the contribution to FLORIS, α_{VAWT} and β_{VAWT} are both set to 0.5 to imitate the existing code. However, in this work, we choose $\alpha_{VAWT} = \beta_{VAWT} = 1$ to sample U_{in} over the full cross-section of the vertical-axis turbine (see figure 3.1 right, where the ratios are just below 1 for illustration purposes). The exact version of FLORIS used in this work can be found in the author's fork [15].

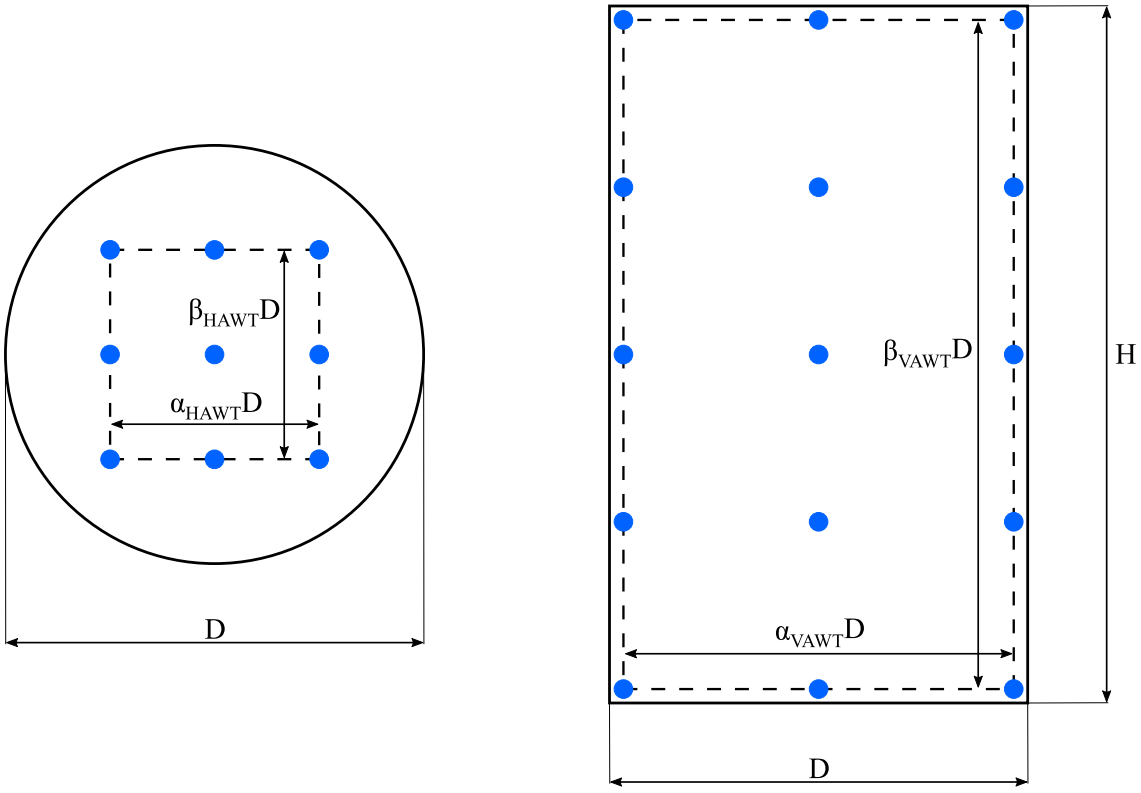


Figure 3.1: Two examples of a `TurbineGrid` in FLORIS. The frontal area of a HAWT (left) and a VAWT (right) is outlined. Both turbine have the same diameter for simplicity, and $H = \frac{5}{3}D$. The incoming streamwise velocity is averaged over the blue points to obtain U_{in}.

With these additions to FLORIS, it is possible to calculate the velocity field and the generated power in a vertical-axis wind turbine farm. This is showcased in figure 3.2 for a 5×5 turbine layout.

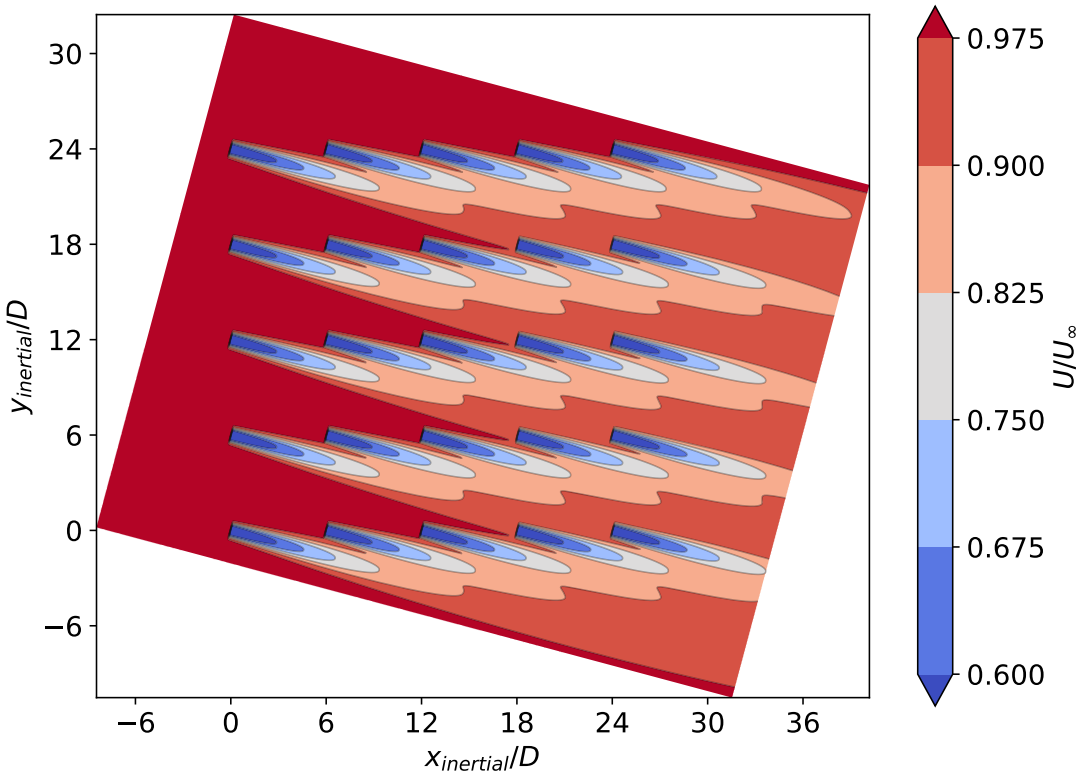


Figure 3.2: Streamwise velocity in a horizontal plane at hub-height in a VAWT farm. This is a modified version of the `small_grid_rotation` test case in FLORIS, where: the height-to-diameter ratio of the rotor is about 2, the freestream velocity is $U_\infty = 8$ m/s, C_T is assigned a constant value of 0.64, the wind direction is 285° (where direction 0° is defined as north to south and the angle is taken clockwise), turbulence intensity is 10 %, and the super-Gaussian model is used. Coordinate system $(x_{inertial}, y_{inertial})$ does not rotate with the wind direction and is used to define the turbine layout. Note that the coordinate system (x, y) used by the velocity model (3.1) does rotate with the wind direction such that x is always in the streamwise direction.

In the example shown in figure 3.2, the velocity field and turbine powers are calculated in a few seconds. FLORIS also makes it possible to evaluate multiple wind directions and wind speeds simultaneously, as well as to optimize the turbine layout given a set of site conditions. The big question is how accurate the results are for vertical-axis turbines.

The accuracy of the results relies heavily on how well the wake model predicts the wake of an isolated turbine and the choice of combination model. Since the super-

Gaussian wake model has shown good performance in six single-turbines cases [6], the focus of this study is on the combination model/wake superposition.

3.1.3 Combination models

Two combination models are used in this work: *freestream linear* and *freestream sos* (sum-of-squares). First, the velocity deficit field behind turbine i given by the super-Gaussian model is denoted as

$$\frac{\Delta U_i(\boldsymbol{x})}{U_\infty} = \frac{U_\infty - U_i(\boldsymbol{x})}{U_\infty} \tag{3.3}$$

where $\boldsymbol{x} = (x, y, z)$.

From (3.3) is it clear that the velocity field $U_i(\boldsymbol{x})$ can be obtained from the wake model. The combined velocity field $U(\boldsymbol{x})$ is then calculated with the following combination models

$$\text{freestream linear:} \qquad U(\boldsymbol{x}) = U_\infty - \sum_{i=0}^{N-1} (U_\infty - U_i(\boldsymbol{x}))$$

$$\text{freestream sos:} \qquad U(\boldsymbol{x}) = U_\infty - \sqrt{\sum_{i=0}^{N-1} (U_\infty - U_i(\boldsymbol{x}))^2}$$

where N is the number of turbines in the wind farm.

The models above are called *freestream models* since the inflow to each turbine is assumed to be equal to the freestream velocity. There exist models where the local inflow to each turbine is taken into account [16]. However, these have not been implemented in FLORIS and are thus excluded from this study. Nevertheless, we acknowledge that Huang obtained good results with the *freestream sos* model on a two-turbine layout when the turbine blades had zero pitch [17].

3.2 General CFD setup

Two-dimensional (2d) CFD simulations are used to evaluate the performance of the combination models used together with the super-Gaussian wake model. The CFD simulations utilize the $k-\varepsilon$ RANS model and the actuator line model (ALM). It is still an open question whether 2d simulations are accurate enough to predict the wake characteristics, C_T, and C_P of a VAWT, given that effects such as tip vortices are neglected when all influence from the vertical direction is excluded. For example, results from blade-resolved 2d RANS simulations by Bianchini [18] matched experimental data well in regards to C_P and the streamwise velocity in the near-wake ($1D$ downstream of the turbine, where D is the turbine diameter). However, the far-wake still needs to be investigated.

The CFD simulations are performed with OpenFOAM v2112 coupled with `turbinesFoam` [19]. The latter is a user-contributed implementation of the ALM.

An introduction to this library can be found in [7]. Despite that the bulk of `turbinesFoam` was developed around 2015-2016, it has been verified that the UNH-RVAT case [7] still produces expected results in OpenFOAM v2112 (but, at the time of writing, a fix [20] is needed to avoid a compilation problem of `turbinesFoam` for this OpenFOAM version). Moreover, the code supports multiple turbines (see e.g. [21]).

3.3 The actuator line model

In the actuator line model (ALM), each turbine blade is modeled as a line which spans the blade length. The line consists of multiple actuator line elements (ALEs). At each such element, the local flow velocity is sampled and then used to calculate the velocity of the air relative to the blade, $\boldsymbol{U}_{rel}$, and the angle of attack. This information is used to obtain element-specific lift and drag coefficients, C_l and C_d, from airfoil polars. Adopting a similar notation as Sørensen [22], the total force $\boldsymbol{F}$ on each ALE (here per unit density) is then given as

$$\boldsymbol{F} = \boldsymbol{F}_l + \boldsymbol{F}_d = \frac{1}{2} U_{rel}^2 cs(C_l \boldsymbol{e}_l + C_d \boldsymbol{e}_d) \tag{3.4}$$

where U_{rel} is the magnitude of the relative velocity, c is the chord length, s is the span of the ALE, and $\boldsymbol{e}_l$ and $\boldsymbol{e}_d$ are lift and drag unit vectors respectively.

An equal and opposite force to $\boldsymbol{F}$ is then applied to the flow, to model the effect of the turbine blade. It is denoted $\boldsymbol{F}_s = -\boldsymbol{F}$, where subscript s indicates that this force is applied as source terms in the momentum equations. As `turbinesFoam` is written for 3d simulations, the force is applied in a region around the ALE by multiplying $\boldsymbol{F}_s$ with a spherical Gaussian function. This function integrates to 1 [23]. In 2d simulations, one may instead use a 2d Gaussian function, which is described in section 3.3.1.

It is important to note that there are different ways to sample the local velocity at the ALEs [23, 24]. Works such as [7, 24] have used linear interpolation of cell values at the location of each ALE, as implemented in `turbinesFoam`. This method should work well "if the actuator line force projection function is symmetric in space" [23, p. 7], as is the case in this work.

3.3.1 Force projection in 2d simulations

In 2d simulations, each turbine blade consists of only one ALE. The force $\boldsymbol{F}_s$ (defined in the previous section) is thus the total blade force that should be applied to the flow. This force is smeared using a 2d Gaussian function $\eta_{\varepsilon,2d}$ which must integrate to 1. The parameter ε [m], called the Gaussian width, controls the width of the region in which the majority of the force is applied. To obtain $\eta_{\varepsilon,2d}$, suppose that an ALE is located at $\boldsymbol{x} = (0, 0, 0)$ and consider the following problem:

$$\text{Find } C \text{ in } \quad \eta_{\varepsilon,2d}(\boldsymbol{x}) = Ce^{-\frac{|\boldsymbol{x}|^2}{\varepsilon^2}} \quad \text{such that} \quad \int_{\mathbb{R}^2} \eta_{\varepsilon,2d}(\boldsymbol{x})\, dA = 1 \tag{3.5}$$

where C is a constant.

The solution (see details in appendix C) is

$$\eta_{\varepsilon,2d}(\boldsymbol{x}) = \frac{1}{\pi\varepsilon^2} e^{-\frac{|\boldsymbol{x}|^2}{\varepsilon^2}}$$

In a pure 2d simulation, the calculations can be done per unit blade span by setting $s = 1$ m and otherwise ignoring the third dimension. Then, the smeared force field can be written as

$$\boldsymbol{f}_{\varepsilon,2d}(\boldsymbol{x}) = \boldsymbol{F}_s \eta_{\varepsilon,2d}(\boldsymbol{x} - \boldsymbol{x}_{ALE})$$

where $\boldsymbol{x}_{ALE}$ is the position of the ALE.

Constructing $\eta_{\varepsilon,2d}$ such that is integrates to 1 ensures that the area integral of $\boldsymbol{f}_{\varepsilon,2d}(\boldsymbol{x})$ over $\mathbb{R}^2$ indeed equals to $\boldsymbol{F}_s$.

However, in OpenFOAM, a 2d simulation is treated much like a 3d simulation. In a 2d case, there is still a domain thickness Δz and the blade span needs to be set to $s = \Delta z$ when calculating the force $\boldsymbol{F}_s$ to be applied. Preferably $\Delta z = 1$ m is chosen such that the force applied in the simulation is per unit blade span. Nevertheless, we here assume that the domain thickness is any positive value for robustness. Further, for implementation purposes, the smeared forcefield needs to be volumetric even in a 2d simulation. Therefore, `turbinesFoam` is modified [25] to use the following forcefield in the case of 2d simulations

$$\boldsymbol{f}_{\varepsilon,OF2d}(\boldsymbol{x}) = \boldsymbol{F}_s \frac{\eta_{\varepsilon,2d}(\boldsymbol{x} - \boldsymbol{x}_{ALE})}{\Delta z}$$

where OF stands for OpenFOAM.

OpenFOAM then adds the source term $\boldsymbol{f}_{\varepsilon,OF2d}(\boldsymbol{x}_i)\Delta\mathcal{V}_i$ to the momentum equation of each cell i, where $\boldsymbol{x}_i$ is the position of the cell center of cell i and $\Delta\mathcal{V}_i$ is the cell volume. Let us ensure that the sum of the above source term over all cells is approximately equal to the full force $\boldsymbol{F}_s$ that the blade exerts on the flow. More formally, let $\Omega_{OF2d} \subset \mathbb{R}^3$ be an infinitely large 2d domain in OpenFOAM. Then we need to show that

$$\sum_{\substack{i \in \text{all cells} \\ \text{in } \Omega_{OF2d}}} \boldsymbol{f}_{\varepsilon,OF2d}(\boldsymbol{x}_i)\Delta\mathcal{V}_i \approx \boldsymbol{F}_s$$

This is done as follows

$$\boldsymbol{F}_s = \int_{\mathbb{R}^2} \boldsymbol{F}_s \eta_{\varepsilon,2d}(\boldsymbol{x} - \boldsymbol{x}_{ALE})\, dA = \sum_{\substack{i \in \text{all cells} \\ \text{in } \mathbb{R}^2}} \int_{\text{cell } i} \boldsymbol{F}_s \eta_{\varepsilon,2d}(\boldsymbol{x} - \boldsymbol{x}_{ALE})\, dA \approx$$

$$\sum_{\substack{i \in \text{all cells} \\ \text{in } \mathbb{R}^2}} \boldsymbol{F}_s \eta_{\varepsilon,2d}(\boldsymbol{x}_i - \boldsymbol{x}_{ALE})\Delta A_i = \sum_{\substack{i \in \text{all cells} \\ \text{in } \Omega_{OF2d}}} \boldsymbol{F}_s \frac{\eta_{\varepsilon,2d}(\boldsymbol{x}_i - \boldsymbol{x}_{ALE})}{\Delta z}\Delta\mathcal{V}_i =$$

$$\sum_{\substack{i \in \text{all cells} \\ \text{in } \Omega_{OF2d}}} \boldsymbol{f}_{\varepsilon,OF2d}(\boldsymbol{x}_i)\Delta\mathcal{V}_i$$

where the integral over cell i is approximated by assuming that the integrand is constant within the cell volume, and ΔA_i is the area of cell i in the xy-plane.

For future reference, 63 % of the blade force is applied within a radius of ε from $\boldsymbol{x}_{ALE}$, 98.2 % within a radius of 2ε, and 99.99 % withing a radius of 3ε.

3.4 Single-turbine CFD validation

3.4.1 Reference data

A single-turbine 3d LES + ALM simulation by Abkar [26] is used as a reference. The framework used by Abkar has been validated [9] against experimental data from Brochier et al. considering the streamwise velocity in the wake of a lab-scale VAWT. Defining x as the streamwise location behind the turbine, and D as the turbine diameter, the validation was performed for $x \leq 5.8D$. From the perspective of a wind farm, $5D$ behind a turbine may be seen as the end of the near-wake region and the beginning of the far-wake.

In Abkar's work, a real-life 200 kW turbine with $D = 26$ m is modeled (called the T1-turbine [27]). The three blades have a NACA0018 profile. Abkar presents two simulations with the original blade length of $H = 24$ m and one in which the blades are extended to $H = 48$ m. The latter is thought to better correspond to the 2d simulations in this work (where the turbine blades have infinite length) and is thus chosen as a reference.

Since the super-Gaussian wake model has already been validated against the reference data [6], we want to assert that the CFD simulations of this work (2d RANS + ALM) yield a similar result before considering a two-turbine layout. Key simulation inputs are shown in table 3.1. Here, c is the chord length, U_∞ is the freestream velocity at hub-height, I_u is the streamwise turbulence intensity at hub-height, λ is the tip-speed ratio, C_T is the turbine thrust coefficient (sometimes called the drag coefficient), and Re_c is the chord-Reynolds number based on the relative velocity magnitude U_{rel}. An estimate of the Reynolds number is used to select adequate airfoil data sets for use in the ALM.

Further, the properties of atmospheric air at sea-level are taken as: pressure $p = 101350$ Pa, density $\rho = 1.2255$ kg/m^3, kinematic viscosity (calculated with the power law) $\nu = 1.45 \cdot 10^{-5}$ m^2/s [28]. The Reynolds number is then given by

$$\mathrm{Re}_c = \frac{cU_{rel}}{\nu} \approx \frac{c\lambda U_\infty}{\nu} \tag{3.6}$$

Table 3.1: VAWT geometry and operating conditions in the reference simulation.

D [m]	H [m]	c [m]	U_∞ [m/s]	I_u	λ	C_T	Re_c
26	48	0.75	7	0.091	3.8	0.64	$1.4 \cdot 10^6$

3.4.2 CFD setup

3.4.2.1 Domain

The computational domain is $60D$ long and $40D$ wide (see figure 3.3). Balduzzi et al. found that the difference in the turbine power coefficient C_P between the chosen domain size and a much larger size ($140D \times 60D$) was only 1 % [29]. All cells have an aspect ratio of 1 and the cell-size of the background mesh is chosen to $\Delta x = \Delta y = 0.2D$. This implies that the number of cells per turbine diameter is 5 the background mesh, 10 in the first refinement box, and 20 in the second refinement box which covers the turbine. This is comparable to Bachant et al., Abkar & Dabiri, and Shamsoddin & Porté-Agel who had 26, 10, and 15 cells per turbine diameter in the turbine region respectively [7, 9, 30]. The mesh has about 120 000 cells.

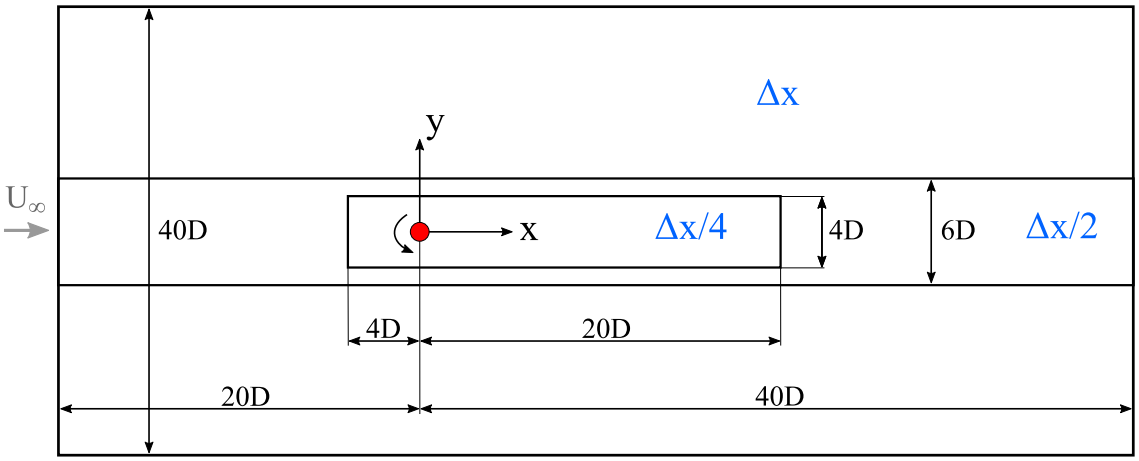


Figure 3.3: Sketch of the domain including the two refinement boxes. The cell-size in each region is indicated in blue. The turbine (shown in red) rotates counter-clockwise and the direction of the flow is from left to right. Everything is drawn to scale except for the domain width ($40D$).

3.4.2.2 Boundary conditions

A uniform velocity U_∞ is used at the inlet, and atmospheric pressure is prescribed at the outlet. In the cross-stream direction, the boundary is placed far away from the turbine and a periodic condition is applied (same as in [9, 30], and similar to the widely used symmetry condition [29]).

For a given turbulence intensity I, the inlet conditions for the $k-\varepsilon$ model are taken as [31]

$$k = \frac{3}{2}(U_\infty I)^2, \quad \varepsilon_{turb} = C_\mu^{0.75}\frac{k^{1.5}}{l} \tag{3.7}$$

where subscript *turb* is used to differentiate the rate of turbulent dissipation from the Gaussian width in the ALM, and $C_\mu = 0.09$. The turbulence length scale is taken as $l = 0.08D$ [31, p. 70], where D is assumed to be a characteristic wake

width. We also assume that the turbulent fluctuations are the same in all directions such that I can be set equal to the known streamwise turbulence intensity I_u (see table 3.1).

There is however a known problem of *turbulence decay* in the $k-\varepsilon$ and $k-\omega$ *SST* model [32], meaning that k (and therefore also the turbulence intensity) decreases in the streamwise direction such that the value at the turbine location may be significantly lower than the value prescribed at the inlet. There are modified turbulence models that are supposed to fix this problem [32], but since no such model is found in OpenFOAM, and implementing it is beyond the scope of this thesis, we resort to study the sensitivity of the ε_{turb} inlet condition.

A note on the turbulent viscosity $\nu_t = C_\mu k^2/\varepsilon_{turb}$ is in order. This property defines "how efficient the turbulence transports (by diffusion) momentum" [33, p. 131]. Condition (3.7) gives $\nu_t/\nu \approx 6 \cdot 10^4$ which is very high compared to a general rule of thumb $\nu_t/\nu \approx 10$ mentioned by Nilsson [34]. As an alternative condition, the turbulence transport may be controlled by enforcing a certain value of ν_t/ν (such that ν_t is known). Then, the inlet value of ε_{turb} can be calculated as

$$\varepsilon_{turb} = C_\mu \frac{k^2}{\nu_t} \tag{3.8}$$

The results of this study show that the turbulence decay increases with ε_{turb} (see section 4.2.1.2). Therefore, a compromise is needed between not too high levels of turbulence transport (ν_t low enough) and not too rapid turbulence decay (ν_t high enough). It turns out that using condition (3.8) with a high value of $\nu_t/\nu = 2000$ still gives a too high ε_{turb}; the turbulence intensity of 9.1 % at the inlet decays to only 0.5 % at the turbine position. Such low turbulence levels are unacceptable, which is why condition (3.7) is used instead. We then get a turbulence intensity of 3.7 % at the turbine.

3.4.2.3 Turbine and ALM setup

The turbine is modeled as three blades, excluding the tower and struts. As mentioned previously, airfoil lift and drag coefficients are needed in the ALM. These are generally a function of the angle of attack α and the chord-Reynolds number Re_c. The coefficients are taken from Sheldahl & Klimas [35], whose NACA0018 dataset covers all α for $\mathrm{Re}_c \in [4 \cdot 10^4, 5 \cdot 10^6]$. Note that Sheldahl & Klimas assumed that the coefficients were Reynolds-independent for $|\alpha| \geq 30°$ for this particular profile. In addition, `turbinesFoam` uses the moment coefficient to account for the pitching moment's contribution to the turbine torque (which subsequently is used to calculate the turbine power). Moment coefficient data is more scarce, so we use the data available for the NACA0015 profile for $\mathrm{Re}_c = \{3.6 \cdot 10^5, 6.8 \cdot 10^5\}$ in the same way as Bachant [7].

In `turbinesFoam`, the dynamic stall model `LeishmanBeddoesSGC` is used with NACA0018-specific coefficients taken from Dyachuk et al. [36, Table 1]. The flow curvature model `Goude` is also used, and the end effects model (or tip loss model) is

disabled since this is a 2d simulation. An overview of the aerodynamic submodels in `turbinesFoam` can be found in [7]. Further, the time step Δt is chosen such that the actuator line moves less than one cell per time step [7] (multiplying with 0.75 to add a safety factor):

$$\Delta t = 0.75\Delta t_{crit} = 0.75\frac{\Delta x_{turbine}}{\lambda U_\infty} \tag{3.9}$$

where $\Delta x_{turbine}$ is the cell-size in the turbine region (where all actuator line elements are located).

Assuming a cell aspect ratio of 1, we can write the standard `turbinesFoam` condition for choosing the Gaussian width, in a similar way as [8], as

$$\varepsilon = \max\left(4\Delta x_{turbine},\ \frac{c}{4},\ \frac{cC_d}{2}\right) \tag{3.10}$$

where C_d is the drag coefficient obtained from airfoil polars by each ALE separately. If the rightmost argument is the largest, then ε may vary between ALEs.

In this work, the ability to switch between the standard condition (3.10) and a fixed value of ε is implemented to facilitate a sensitivity analysis. Based on this analysis, a fixed value of $\varepsilon = 2.6$ m was chosen (see section 3.4.3.1).

The additions to `turbinesFoam` [25] can now be summarized by providing a typical `forceProjection` dictionary (see listing 1) which is added as an entry in the `crossFlowTurbineALSourceCoeffs` dictionary. The latter resides in the `fvOptions` dictionary in OpenFOAM and is used to define the turbine.

```
forceProjection
{
    forceProjectionModel  Gaussian2D; // Gaussian2D || Gaussian3D
    domainThickness       1;          // Only needed in Gaussian2D
    epsilonMethod         Fixed;      // Standard || Fixed
    epsilonFixed          2.6;        // Only needed in Fixed method
}
```

Listing 1: Dictionary `forceProjection` added to `fvOptions`.

3.4.2.4 Solvers

The PIMPLE solver is set to operate in PISO-mode, which effectively means that the PISO algorithm is used to solve the discretized equations. The second-order scheme `backward` is used in time. Regarding convection schemes, the second-order scheme `Gauss linearUpwind` is used for velocity, and `Gauss limitedLinear 1` is used for turbulence properties. The latter is a `linear` (central differencing) scheme that "limits towards `upwind` in regions of rapidly changing gradient" [37]. Thus, it is a combination of a second-order and first-order scheme.

The velocity and pressure field are solved to a tolerance of 10^{-6}. Turbulence fields are solved to 10^{-8} to ensure enough iterations in the linear solvers (more than 1 iteration is often needed in the beginning of the simulation).

3.4.2.5 Convergence

The turbine power coefficient is usually monitored to determine simulation convergence. However, the average C_P over a revolution changes less than 0.1 % between revolutions already at $t = 69$ s. For comparison, it takes 223 s for a particle to travel through the whole domain, so it is not surprising that the wake is still undeveloped at this time. Therefore, the convergence metric of this study is based on a velocity probe at $x/D = 12$ to make sure that the far-wake is fully developed.

Two types of averaging are used to process the velocity probe data and ultimately calculate the convergence metric δ. First, an attempt is made to average out the unsteady effects. This is done by dividing the data into groups in time, separated by an averaging time step $(\Delta t)_{avg}$ equal to two turbine periods, and then calculating the average velocity deficit $\Delta U_{avg,j}$ in each group j. The corresponding times are $t_j = j(\Delta t)_{avg} + (\Delta t)_{avg}/2$, where $j \geq 0$ is an integer. Then, at a particular time t_i, the change in $\Delta U_{avg,j}$ between consecutive groups is averaged over the last $N = 20$ groups to determine the convergence criterion as

$$\delta_i = \frac{1}{N} \sum_{j=i-N+1}^{i} \frac{|\Delta U_{avg,j} - \Delta U_{avg,j-1}|}{\Delta U_{avg,j-1}} < 0.1\%$$

The purpose of the second averaging is to avoid erroneous convergence at local maxima or minima. After convergence, the velocity field and turbulent kinetic energy field are averaged over 20 turbine revolutions before terminating the simulation.

3.4.3 CFD sensitivity analysis

The focus of the sensitivity analysis is on the cell-size Δx of the background mesh, the Gaussian width ε, and the inlet condition of ε_{turb} in the turbulence model. The first two parameters are related and are thus considered in a joint analysis.

3.4.3.1 Sensitivity of cell-size and Gaussian width

In the present study, the mesh-based ε-condition in (3.10) is the most critical, meaning that $\varepsilon = 4\Delta x_{turbine}$ for all ALEs at all time steps (recall that $\Delta x_{turbine} = \Delta x/4$ since the mesh is refined in the turbine region). This condition ensures that the actuator line forces are spread over enough cells and it is thus clear that the Gaussian width depends on the cell size. Other studies have conducted mesh-convergence studies while maintaining condition (3.10) meaning that, if the mesh-based condition is critical, then both Δx and ε are reduced when the mesh is refined. Here, we instead check the sensitivity of each parameter separately.

For the simplicity of integer values, the number of cells in the x-direction in the background mesh n_x (hereafter called the mesh-size) is used to represent to size

of the cells instead of Δx (which is equal to L/n_x). A starting point $n_x = 300$ and $\varepsilon = 5.2$ m is selected for the analysis, since this gives a reasonable number of cells per turbine diameter in the turbine region (namely 20) and follows the rule of $\varepsilon = 4\Delta x_{turbine}$. We then try to spread the blade forces over a smaller area by choosing $\varepsilon = 2.6$ and 1.3 m; closer to the chord length $c = 0.75$ m. The velocity in the wake is evaluated for at least three different mesh-sizes for each ε.

To quantify the sensitivity of n_x for a particular ε, centerline velocity deficit profiles are compared. The profile of every n_x, denoted $\Delta U(x, 0)$, is compared to the profile of the largest n_x, denoted $\Delta U_{ref}(x, 0)$. Velocity is sampled at $N = 100$ points using the uniform partition $1D = x_0 < x_1 < \ldots < x_i < \ldots < x_{N-1} = 12D$. The root-mean-square difference between a profile and its reference can then be calculated as

$$d_{RMS} = \frac{1}{(\Delta U)_{char}} \sqrt{\frac{1}{N} \sum_{i=1}^{N} \left[\Delta U(x_i, 0) - \Delta U_{ref}(x_i, 0)\right]^2}$$

where $(\Delta U)_{char}$ is a characteristic velocity deficit used for normalization. It is taken as the mean value of the reference velocity profile.

The sensitivity of ε is evaluated in a similar way. However, we only use a single, large enough, n_x to compare velocity profiles between $\varepsilon = 5.2$, 2.6, and 1.3 m. The lowest ε is used as a reference.

Due to the chosen time step criterion (3.9), it should be noted that Δt is proportional to Δx, meaning that finer grids have smaller time steps. After choosing a suitable Δx and ε from the sensitivity study above, a time step twice as small as that given by (3.9) is evaluated, to assert insensitivity of Δt.

3.4.3.2 Sensitivity of turbulence model inlet conditions

Regarding the setup of the turbulence model, we mainly focus on the dissipation rate of turbulent kinetic energy at the inlet $\varepsilon_{turb,in}$. Let $\varepsilon_{turb,in,standard} = 0.0375$ m^2/s^3 be the value given by inlet condition (3.7); hereafter called the standard condition (hence the *standard* subscript). This is taken as an upper bound since turbulence decay is severe at this point and a test with a higher value $\varepsilon_{turb} = 1.15$ (such that $\nu_t/\nu = 2000$) gave unrealistic results, as mentioned in section 3.4.2.2. Consequently, lower values than $\varepsilon_{turb,in,standard}$ are evaluated.

The idea of lowering $\varepsilon_{turb,in}$ to decrease turbulence decay was inspired by Monni [38]. However, it must be emphasized that ν_t/ν is already in the order of 10^4 when the inlet value is $\varepsilon_{turb,in,standard}$, and the ratio becomes even larger for the $\varepsilon_{turb,in}$ tested in this sensitivity analysis. It is therefore likely that the diffusion of momentum due to turbulence becomes too high at some point. Nevertheless, this study should give an indication of how sensitive the simulations are to the turbulence inlet conditions.

For convenience, a new variable is introduced

$$\varepsilon^*_{turb,in} = \frac{\varepsilon_{turb,in}}{\varepsilon_{turb,in,standard}}$$

In the first part of the study, the original domain (see figure 3.3) is modified by reducing the width to $W = 1D$ and removing the turbine. In this way, k can be sampled along the center of the domain, from the inlet up to $10D$ behind the turbine, to evaluate the turbulence decay. This is done for $\varepsilon^*_{turb,in} \in \{1, 0.5, 0.1, 0.025\}$.

In the second part of the study, the same values of $\varepsilon^*_{turb,in}$ are used but the simulations are done in the original domain where the turbine is included. The wake velocity field is compared between the four cases.

3.4.4 Summary of chosen CFD setup

Parameters for the single-turbine CFD simulation are chosen based on the results of the sensitivity analysis (see section 4.2.1). The parameters are summarized in table 3.2 for convenience.

Table 3.2: CFD parameters.

Parameter	Value
Domain size $L \times W$	$60D \times 40D$
Background mesh $n_x \times n_y$	300×200
Time step Δt	0.037 s based on (3.9)
Simulation end time	446 s
Gaussian width ε	2.6 m
Inlet turbulence intensity	0.091
Incoming turbulence intensity to the turbine	0.037
Inlet ε_{turb}	0.0375 m^2/s^3 based on (3.7)
Compressibility	Incompressible

3.5 Two-turbine study

The goal of this section is to outline how CFD simulations and wake model results are compared on two-turbine layouts. First, the layouts are presented and an important measurement region is defined. Details are then given for the CFD setup and wake model setup.

3.5.1 Turbine arrays and scope

The turbine arrays considered in this study are shown in figure 3.4. The naming convention of the arrays is *pairX*, meaning a pair of turbines rotated X degrees about the center of the first turbine T0. A turbine spacing as large as $11D$ has been

observed in literature (examples of layouts can be found in [4, 39]). However, a rather small spacing of $5D$ is chosen in this work to ensure significant wake interaction.

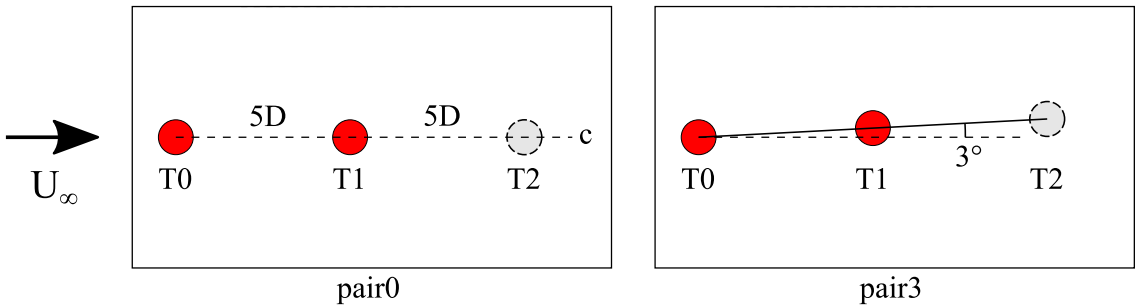


Figure 3.4: Two array layouts, where red circles indicate VAWTs and grey circles indicate a hypothetical third turbine. Turbine i is denoted Ti and the centerline is denoted c. All turbines rotate counter-clockwise. The sketches are drawn to scale except for the domain size.

The *pair0* layout is a worst-case scenario in a wind farm since downstream turbines are completely waked by upstream turbines, resulting in low inflow velocities to each turbine and subsequently low power production. The *pair3* layout is equivalent to a 3° change of the wind direction compared to the *pair0* layout (but here the turbine positions are rotated instead of the wind direction to make the CFD setup easier).

This study is largely focused on how the wakes of several turbines interact. We want to know how the wake model results are affected by the combination model and how the results compare with CFD simulations. Therefore, the velocity field is analyzed behind T1 (the second turbine) where the combination model is active.

The wake model can be used to predict the inflow velocity $U_{in,i}$ to turbine i. Practically, a `TurbineGrid` is created which contains multiple grids in the yz-plane that cover the frontal area of every turbine (see figure 3.1). At each turbine i, the wake model returns the velocity at the grid points and these velocities are averaged to obtain $U_{in,i}$. To make the comparison more fair between the wake model and the 2d CFD simulations of this work, the velocities used to calculate $U_{in,i}$ are only sampled at hub-height in both cases. More specifically, 51 equally spaced sample points are taken from the set $\{(x, y) : x = x_i,\ |y - y_i| \leq 0.5D\}$, where (x_i, y_i) is the location of the turbine hub. In FLORIS, this is done by specifying the number of points in the cross-stream and vertical direction of the `TurbineGrid` as 51×1. For a given $C_{P,i}$, the turbine power P_i predicted by the wake model can then be calculated as

$$P_i = \frac{1}{2}\rho A U_{in,i}^3 C_{P,i} \tag{3.11}$$

Note that P_0 is simply obtained by inserting $U_{in,0} = U_\infty$ in (3.11). Further, the inflow velocity $U_{in,1}$ used to calculate P_1 can be predicted with the super-Gaussian model alone (no combination model is needed). However, the predicted power P_2 of a hypothetical third turbine T2 depends on the combination model and is thus compared between the different cases in this study.

3.5.2 CFD setup

The two-turbine CFD simulations are set up by adding an identical second turbine to the single-turbine validation case detailed in section 3.4. We need to make a distinction between the classical tip-speed ratio (hereafter denoted λ_∞), based on the freestream velocity, and a local tip-speed ratio λ, based on the inflow velocity to a specific turbine. For a given turbine i, their definitions are given as

$$\lambda_{\infty,i} = \frac{\omega_i R}{U_\infty}, \quad \lambda_i = \frac{\omega_i R}{U_{in,i}}$$

where R is the turbine radius, and ω_i and $U_{in,i}$ is the rotation speed and inflow velocity of turbine i, respectively. The latter is sampled according to section 3.5.1.

Both turbines are set to operate at the same local tip-speed ratio such that $\lambda_0 = \lambda_1 = \lambda = 3.8$. This value is close to the optimal value $\lambda_{opt} = 4.3$ found experimentally [27] for the reference turbine mention in section 3.4.1. Note that λ_{opt} maximizes the power coefficient $C_p(\lambda)$ for a given inflow velocity. The rotation speed ω_i of each turbine i is set according to

$$\omega_i = \frac{\lambda U_{in,i}}{R} \tag{3.12}$$

Since T1 is in the wake of T0, $U_{in,1} < U_{in,0} = U_\infty$, which implies that $\omega_1 < \omega_0$. In other words, the rotation speed of the waked turbine is adjusted to maintain an optimal (maximum) C_P, which is a realistic control strategy. The inflow velocity $U_{in,1} \approx 4.85$ m/s is sampled to determine ω_1.

Condition (3.12) is practically enforces by setting the `tipSpeedRatio` variable for every turbine in `turbinesFoam`. Note that this variable is based on the freestream velocity and is thus denoted $\lambda_{\infty,i}$. The inputs to `turbinesFoam` are calculated as

$$\lambda_{\infty,i} = \lambda \frac{U_{in,i}}{U_\infty}$$

which gives $\lambda_{\infty,0} = 3.8$ and $\lambda_{\infty,1} = 2.63$.

In this work, the power and thrust coefficient of turbine i are based on the local inflow velocity as

$$C_{P,i} = \frac{P_i}{\frac{1}{2}\rho A U_{in,i}^3}, \quad C_{T,i} = \frac{T_i}{\frac{1}{2}\rho A U_{in,i}^2} \tag{3.13}$$

where T_i is the thrust force of turbine i.

Note that the power and thrust coefficients outputted by `turbinesFoam` are based on the freestream velocity and thus have to be rescaled to obtain $C_{P,i}$ and $C_{T,i}$.

3.5.3 Wake model setup

In the wake model, the power and thrust coefficient are set to those obtained in the CFD simulation; i.e. $C_{P,i}$ and $C_{T,i}$ as defined in (3.13). Their values are given later

in table 4.2. This ensures that if the wake model predicts the same inflow velocity as in the CFD simulation, then the predicted power and thrust force will also be the same. Further, in FLORIS, $C_{P,i}$ and $C_{T,i}$ are assumed to be constant for all wind speeds, which effectively means that the rotation speed of the turbines are adjusted to maintain a certain tip-speed ratio, in the same way as in the CFD simulations.

4

Results & discussion

We begin by showing a proof of concept regarding the 2d force projection implemented in the actuator line model. This technique is then used throughout the study. A single-turbine case is used to validate the CFD simulations and assess the sensitivity of important parameters. Then, a comparison between the super-Gaussian wake model and CFD simulations is performed with a two-turbine layout.

4.1 Force projection in 2d: Proof of concept

Bachant's three-dimensional UNH-RVAT (UNH reference vertical-axis turbine) case [7, 40] is modified to a 2d case by using the force projection presented in section 3.3.1. The case replicates a tow tank experiment with a 1 m diameter turbine. In the simulations, the turbine is stationary while the freestream velocity is set to $U_\infty = 1$ m/s. The solid black line in figure 4.1 shows Bachant's original streamwise velocity profile along a line in the cross-stream direction at hub-height, $1D$ behind the turbine. The same case but with the end effects model (or tip loss model) turned off is also shown. In the *2d* cases, the domain thickness is reduced to one cell, end effects are turned off, and struts are removed. Finally, the width of the domain is increased from about $3.7D$ to $50D$ in the *2d-wide* case.

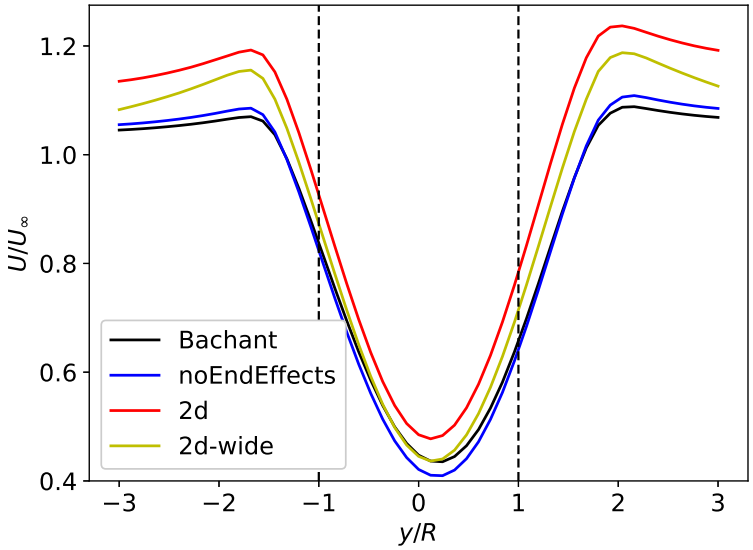


Figure 4.1: Velocity profiles along a line in the cross-stream direction at hub-height, $1D$ behind the turbine. A comparison is made between 3d and 2d simulations of the UNH-RVAT. The turbine radius is denoted by R and the extent of the turbine is shown with dashed lines.

Hereafter, the *Bachant* and *noEndEffect* cases will collectively be called *3d*. Figure 4.1 shows reasonable agreement between the 2d and 3d simulations. The difference is likely not due to the end effects model being turned off, since this model had a low impact in 3d (compare the black and blue lines). Rather, it is probably caused by blockage. One part can be explained by the blockage ratio, which is the turbine cross-section area relative to the domain cross-section area in a yz-plane going through the center of the turbine. This ratio is higher in the *2d* case compared to those in 3d, 27 % versus 11 %, forcing the flow around the turbine and giving rise to the high velocities seen at $|y/R| \approx 2$. However, reducing the blockage ratio to 2 % in the *2d-wide* case does not quite remove this effect, since the flow still cannot go below or above the turbine as in 3d.

4.2 Single-turbine CFD validation

In this section, a comparison between the CFD simulation of this work, the reference simulation by Abkar [26], and the super-Gaussian wake model is first presented. These cases are denoted OF, LES, and FLORIS respectively, where OF stands for OpenFOAM. The setup of the OF simulation is based on the results of the sensitivity analysis detailed in the next section (see table 3.2 for a summary of the setup). Inputs to the wake model are taken from table 3.1.

For each of the three cases, a velocity deficit profile is sampled along the centerline of the domain (see figure 4.2).

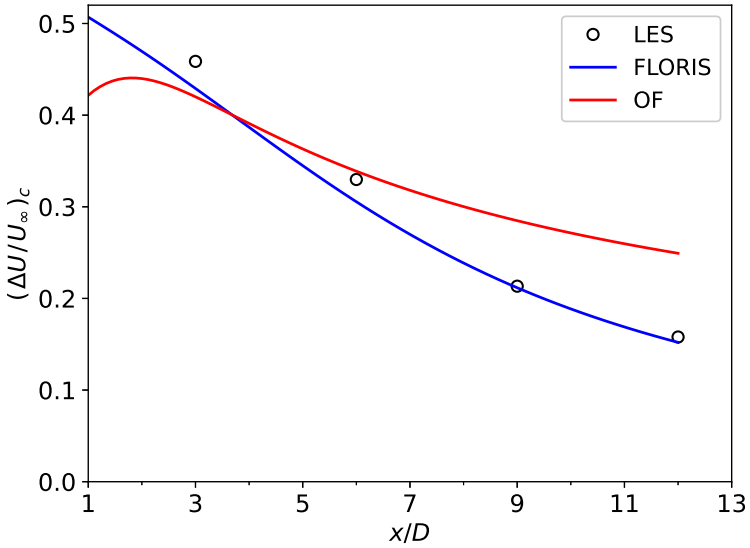


Figure 4.2: Centerline velocity deficit profiles behind a single turbine compared between the OpenFOAM simulation of this work, the LES reference data by Abkar [26], and the FLORIS super-Gaussian wake model.

From figure 4.2 it can be seen the near-wake velocity is similar in all cases. The agreement between the wake model and the LES reference data is excellent, which is partly due to the fact that the reference case was used (along with 5 other cases) to tune the wake model parameters. However, the wake recovery rate is too low in the OF simulation. For example, the relative difference in centerline velocity deficit between the LES and OF simulation is 33 % at $x/D = 9$ and 58 % at $x/D = 12$. A similar result was found by Huang when comparing a 3d RANS + ALM simulation (also utilizing `turbinesFoam` and the $k-\varepsilon$ model) to a wind tunnel experiment [17, sec. 8.3.1]. In that case, the relative difference in $(\Delta U/U_\infty)_c$ between the simulation and the experiment was 89 % at $x/D = 10$ due to a low recovery rate.

It is well-known that a higher turbulence intensity leads to a faster wake recovery for HAWTs [41]. We here assume that the same is true for VAWTs, as is done in the super-Gaussian wake model. The low recovery rate observed above is likely related to the turbulence model and the low turbulence intensity at the turbine (3.7 % instead of the expected 9.1 % due to the turbulence decay mentioned in section 3.4.2.2). How this is affected by the inlet condition of ε_{turb} is investigated in section 4.2.1.2.

There is an uncertainty in comparing the three-dimensional LES and FLORIS case to the two-dimensional OF case. However, limiting the comparison to the velocity in the horizontal plane at hub-height, the 2d force projection test (see section 4.1) only showed a moderate effect of the vertical direction, and the 3d $k-\varepsilon$ simulation by Huang mentioned earlier had a low recovery rate despite including the third dimension. Thus, it is believed that the largest error in this study stems from the turbulence model and not the 2d simplification.

It should be noted that there is an uncertainty in the recovery rate in LES simulations as well. One reason for this is the lack of far-wake experimental data, which limited the validation of Abkar & Dabiri's LES+ALM framework to $x/D \leq 5.8$. In addition, Shamsoddin & Porté-Agel showed that the recovery rate in LES+ALM simulations depends on the LES setup [42].

As a complement to the discussion above, cross-stream velocity profiles are shown for the same cases (see figure 4.3).

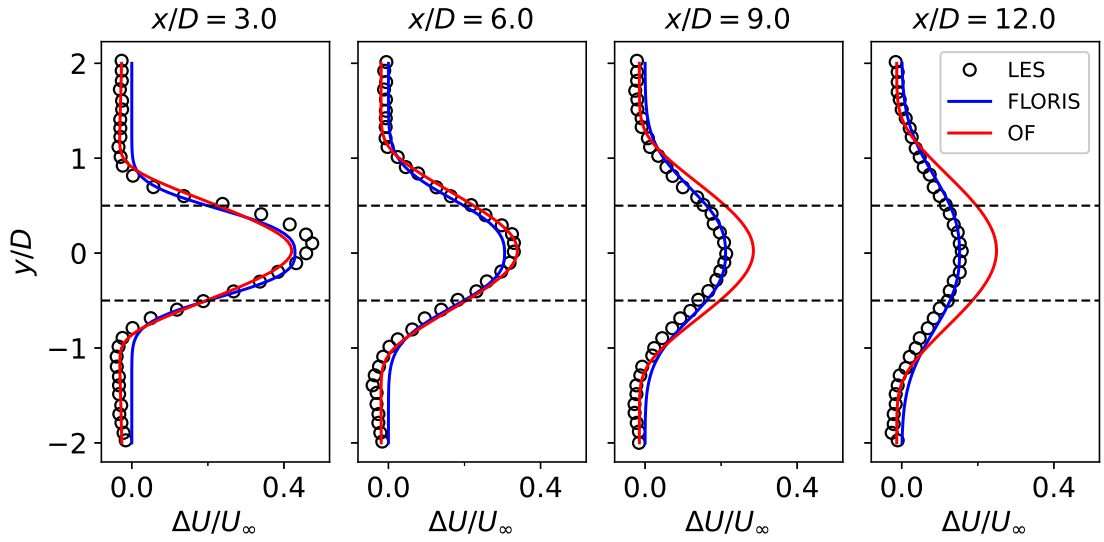


Figure 4.3: Velocity deficit profiles in the cross-stream direction, at several locations downstream of a single turbine. A comparison in made between the OpenFOAM simulation of this work, the LES reference data by Abkar [26] and the FLORIS super-Gaussian wake model. The dashed lines show the extent of the turbine.

Figure 4.3 indicates that the shape of the velocity profile is well-predicted in the OF simulation. However, the larger centerline deficit in the far-wake can clearly be observed.

4.2.1 Sensitivity analysis

4.2.1.1 Sensitivity of cell-size and Gaussian width

The sensitivity of the cell-size and Gaussian width ε is mainly evaluated by looking at centerline velocity deficit profiles. As mentioned previously, the number of cells in the x-direction in the background mesh n_x is used to represent the size of the cells. To maintain a cell aspect ratio of one, we have $n_y = \frac{W}{L} n_x = \frac{2}{3} n_x$. The simulations show some sensitivity to ε (see figure 4.4), while the sensitivity to n_x is very small.

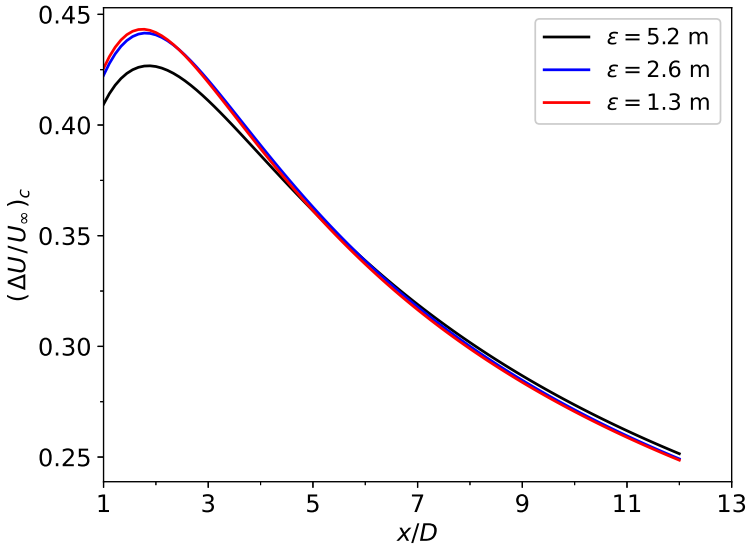


Figure 4.4: Centerline velocity deficit profiles showing the sensitivity of the Gaussian width in the CFD simulation for mesh-size $n_x = 600$.

To quantify the sensitivity, a centerline velocity deficit profile is sampled for multiple combinations of (ε, n_x). The difference d_{RMS} (defined in 3.4.3.1) between a given profile and a reference profile is shown in table 4.1. For example, in the first part of the table where $\varepsilon = 5.2$ m, the profiles for $n_x \in \{120, 180, 240, 300\}$ are compared to the profile of $n_x = 600$; which is assumed to be the most correct for this particular Gaussian width. Similar comparisons are made in the second and third part of the table. In the last part, the sensitivity of ε seen in figure 4.4 is quantified.

Table 4.1: For different Gaussian widths ε and mesh-sizes n_x, the difference d_{RMS} between the obtained velocity profile and a reference profile is shown. Each part of the table has its own reference profile. The ε and n_x chosen in the following simulations are emphasized in bold. The most significant difference is highlighted in red.

ε [m]	n_x	d_{RMS}	$\varepsilon/\Delta x_{turbine}$
5.2	120	0.0071	1.6
5.2	180	0.0030	2.4
5.2	240	0.0016	3.2
5.2	300	0.0010	4.0
5.2	600	reference	8.0
2.6	240	0.0020	1.6
2.6	**300**	0.0011	2.0
2.6	480	0.0002	3.2
2.6	600	reference	4.0
1.3	480	0.0017	1.6
1.3	600	0.0008	2.0
1.3	1200	reference	4.0
5.2	600	0.0203	8.0
2.6	600	0.0040	4.0
1.3	600	reference	2.0

From the data above, the largest difference (about 2 % in d_{RMS}) is found in the last part of the table between $\varepsilon = 5.2$ and 1.3 m. A similar sensitivity to Gaussian width can likely be found for other mesh-sizes than $n_x = 600$. A larger d_{RMS} is also found for (ε, n_x) = (5.2 m, 120) where the mesh is coarse and the force is spread over few cells ($\varepsilon/\Delta x_{turbine} = 1.6$ while the standard condition is 4). One might think that all cases of $\varepsilon/\Delta x_{turbine} = 1.6$ would show larger d_{RMS}, but the data indicates that, if the mesh is fine, then the force can be spread over few cells without any significant loss of accuracy in the velocity field.

To get a better sense of the quantity $\varepsilon/\Delta x_{turbine}$, turbine blade forces (per unit volume) are shown in figure 4.5. Recall that each of the three turbine blades are represented by one actuator line element that exerts the blade forces on the flow within a circular region around the blade's current position. Regions in which the blade force magnitudes are less than 0.1 % of the global max value are omitted (gray in the figure). Since ε is constant in this example, the forces are spread over twice as many cells when $n_x = 600$ compared to $n_x = 300$. This is also indicated by the ratio $\varepsilon/\Delta x_{turbine}$ which is 4 in the former case and 2 in the latter.

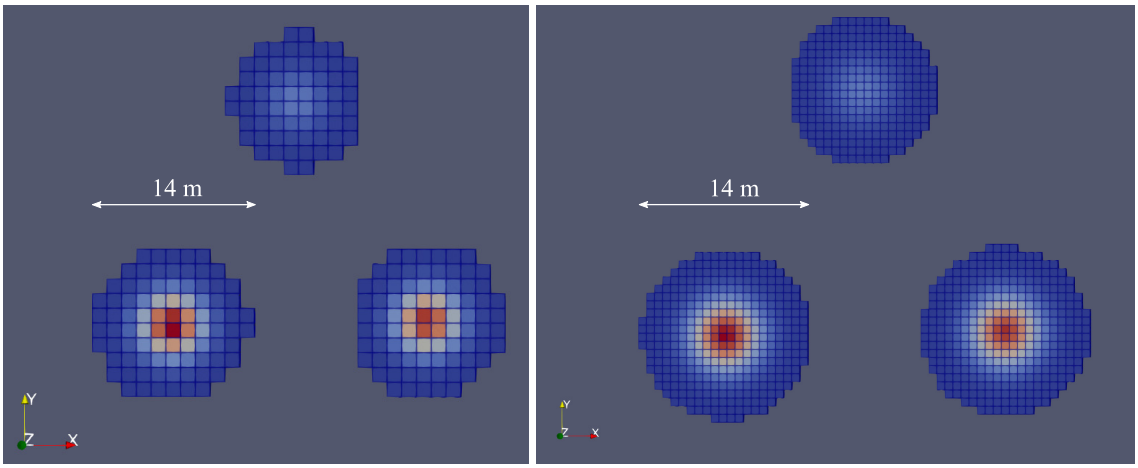


Figure 4.5: Magnitude of instantaneous blade forces (per unit volume) applied by the actuator line method with $\varepsilon = 2.6$ m. Left: $n_x = 300$. Right: $n_x = 600$. Red means high force and blue means low force.

From figure 4.5, it seems reasonable to spread the force over at least as many cells as in the left pane. This translates to the following condition, which is also a safe choice according to table 4.1:

$$\varepsilon/\Delta x_{turbine} \geq 2 \tag{4.1}$$

More specifically, $(\varepsilon, n_x) = (2.6 \text{ m}, 300)$ is chosen in the following simulations unless otherwise noted. For this combination, the results are converged with respect to ε (no need to make is smaller according to the last part of table 4.1), condition (4.1) is fulfilled, and a rather coarse mesh can be used (no need for a finer mesh according to the second part of table 4.1) which saves computational time. Note that the computational time scales as n_x^3 since smaller time steps are used for finer grids. For the chosen parameters, the simulation is rerun with a time step size half of that given by condition (3.9) with minimal effects on the results.

Lastly, it's worth mentioning that the sensitivity to ε is larger in terms of turbulent kinetic energy k than the streamwise velocity. This is especially true in the near wake as seen in figure 4.6. Here, k_{in} is the value of k at the inlet given by (3.7).

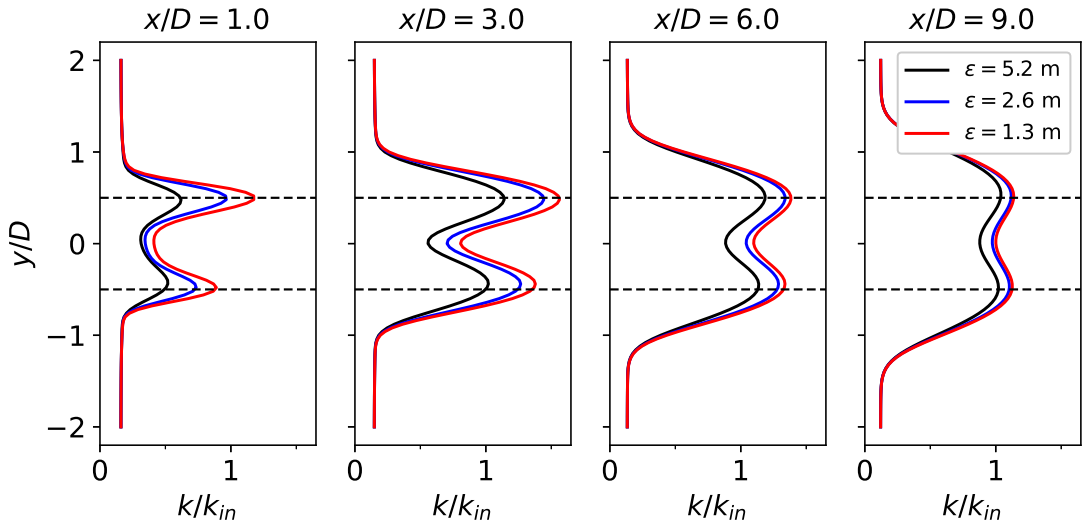


Figure 4.6: Turbulent kinetic energy profiles in the cross-stream direction, at several locations downstream of a single turbine. A comparison is made between different Gaussian widths in the CFD simulation. The mesh-size is $n_x = 600$ and the dashed lines show the extent of the turbine.

The two peaks seen at $|y/D| \approx 0.5$ in figure 4.6 have previously been observed in both experiments and CFD simulations [7, 42]. The difference in k at $x/D = 1$ and $y/D = 0.5$ is 47 % between $\varepsilon = 5.2$ and 1.3 m, and 18 % between $\varepsilon = 2.6$ and 1.3 m. However, since the the profiles of k does converge in the far-wake and the differences in k in the near-wake does not affect the wake velocity much (as shown in the last part of table 4.1), the combination of (ε, n_x) chosen above is kept.

4.2.1.2 Sensitivity of turbulence model inlet conditions

The sensitivity of the inlet condition of the turbulent dissipation rate ε_{turb} is analysed in two parts. In the first part, k is sampled in the streamwise direction in a domain without a turbine (see figure 4.7) for different inlet values $\varepsilon_{turb,in}$. Recall that $\varepsilon^*_{turb,in}$ is the chosen inlet value relative to that given by a standard condition (3.7). The inlet is located at $x/D = -20$ and the turbine would normally be located at $x/D = 0$.

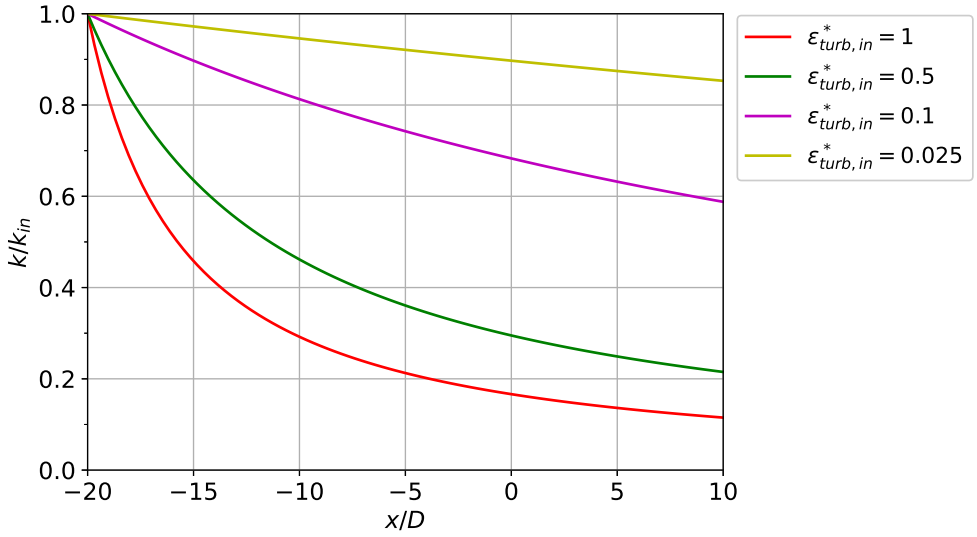


Figure 4.7: Turbulence decay for different inlet values $\varepsilon_{turb,in}$ in the CFD simulation. No turbine is present in the domain.

From figure 4.7, it is clear that a lower value of $\varepsilon_{turb,in}$ gives less turbulence decay. Moving on to the second part of the study, the turbine is inserted and the centerline velocity deficit is sampled behind the turbine. The result is shown in figure 4.8 where the red line almost matches the red line in figure 4.2. The small difference is due to that the standard value of the Gaussian width (5.2 m) for $n_x = 300$ was used in this study instead of 2.6 m chosen in the previous section. This does not matter much though, since the sensitivity to $\varepsilon_{turb,in}$ is much larger than the sensitivity of the Gaussian width.

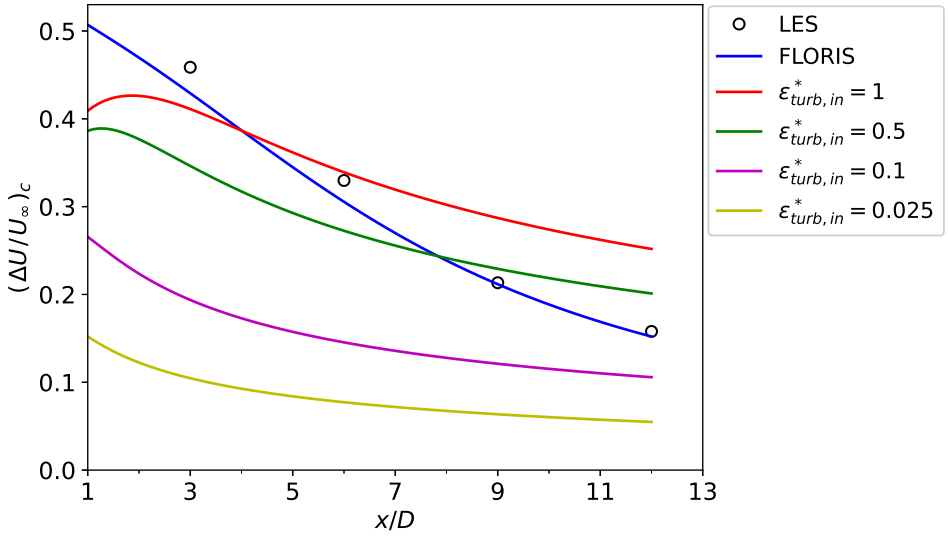


Figure 4.8: Centerline velocity deficit for different inlet values $\varepsilon_{turb,in}$ in the single-turbine CFD simulation. The CFD profiles are compared to the LES reference data by Abkar [26] and the FLORIS super-Gaussian wake model.

Not only are the simulations sensitive to the inlet value of ε_{turb}, but decreasing the value too much also gives an unreasonably low velocity deficit in the wake. This is most likely due to excessive diffusion of momentum caused by a high turbulent viscosity. In conclusion, a good compromise between sustained turbulence levels and a reasonable turbulent viscosity is not found in this study. Therefore, in future research with the actuator line model, LES is seen as an attractive alternative to RANS with the $k-\varepsilon$ model. Nevertheless, we continue the study by choosing $\varepsilon^*_{turb,in} = 1$, i.e. using inlet condition (3.7). This value allows the wake velocity to be well-predicted in the CFD simulation for $x/D \in [3, 7]$ which is an important region in the two-turbine study. For completeness, cross-stream profiles of $\Delta U/U_\infty$ and k from this sensitivity analysis are provided in appendix D.

4.3 Two-turbine study: A comparison between the wake model and CFD simulations

After completing the single-turbine validation, CFD simulations and wake model calculations can be performed on multiple-turbine arrays. First, the CFD results of the *pair0* layout (two turbines in a row) are analyzed and compared to the single-turbine CFD simulation. For the same turbine layout, the wake model velocity field is obtained with two different combination models and compared to the CFD simulation. The same comparison is finally made on a modified layout (*pair3*) which represent a change in wind direction.

4.3.1 Turbine array pair0

4.3.1.1 CFD results

The first turbine layout consists of two turbines that are aligned in the streamwise direction and placed at $x/D = 0$ and $x/D = 5$ respectively. Table 4.2 shows the power and thrust coefficients given by the CFD simulation. The coefficients are almost the same for both turbines, which is expected since their rotation speed is adjusted to maintain a fixed local tip-speed ratio λ. In table 4.2, it can also be seen that value of the thrust coefficients are rather close to the value $C_T = 0.64$ obtained in the LES simulation by Abkar [26].

Table 4.2: Power and thrust coefficients from the CFD simulation.

Turbine index i	$C_{P,i}$	$C_{T,i}$
0	0.359	0.695
1	0.365	0.702

In figure 4.9, a comparison is made between this case (*pair0*) and the singe-turbine case discussed in section 4.2. The left pane shows the velocity deficit along the centerline ($y = 0$). The right pane shows the change of centerline velocity deficit in x per turbine diameter, denoted

$$\frac{d}{dx^*}\left(\frac{\Delta U}{U_\infty}\right)_c = \frac{d}{dx}\left(\frac{\Delta U}{U_\infty}\right)_c \cdot D$$

where $x^* = x/D$.

The derivative above is used to compare the recovery rate of the wake behind the second turbine in the *pair0* case to the recovery rate behind a single turbine.

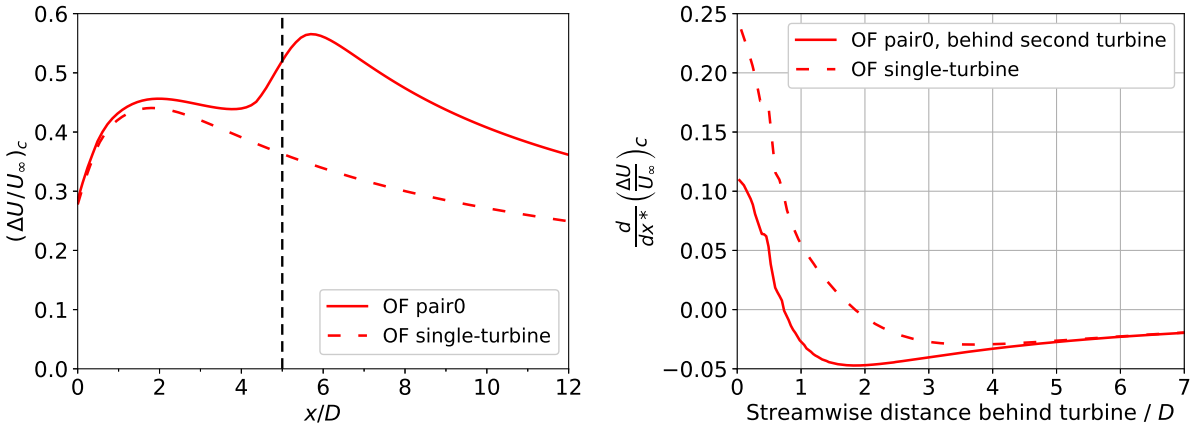


Figure 4.9: Comparison of OpenFOAM (OF) results between the two-turbine *pair0* case and the single-turbine case. Left: Centerline velocity deficit. The dashed line indicates the position of the second turbine (T1) in the *pair0* case. Right: Derivative of centerline velocity deficit. Note that this plot is generated by sampling the centerline velocity for $x/D \in [0, 7]$ in the single-turbine case and for $x/D \in [5, 12]$ in the *pair0* case.

From figure 4.9 left, it is concluded that T1 has an upstream effect on the wake. Specifically, the flow starts to slow down significantly already at $1D$ in front of T1 (at $x/D = 4$). It can also be seen that the velocity deficit in the single-turbine case initially increases before the wake starts to recover. The maximum deficit is located around $x/D = 2$. The same thing is observed behind T1 in the *pair0* case, although the maximum deficit is located less than $1D$ behind the turbine. These maximums can clearly be seen in the right pane of the figure where the derivative is zero. Not only does the wake start to recover earlier behind T1 in the *pair0* case, but the recovery rate is also higher (the derivative is smaller) up until about $5D$ behind the turbine compared to the single-turbine case.

The faster wake recovery of the waked turbine is likely due to that T0 increases the turbulence levels in the wake much above ambient levels (see for example figure 4.6). Indeed, the incoming turbulence intensity to T0 is 3.7 % while the average incoming intensity to T1 is 10.0 %. The latter value is obtained by averaging the turbulent kinetic energy k in the single-turbine case at $x/D = 5$ along a line in the cross-stream direction that spans the turbine cross-section, and calculating the average turbulence intensity as $I_{avg} = (\frac{2}{3}k_{avg})^{1/2}/U_\infty$.

4.3.1.2 Comparison between wake model and CFD results for different combination models

The super-Gaussian model is used together with two combination models, *freestream linear* and *freestream sos* (sum-of-squares), to predict the velocity in the two-turbine *pair0* case. In figure 4.10, the wake model results are compared to the OpenFOAM simulation described in the previous section (now denoted as *OF* instead of *OF pair0* for brevity).

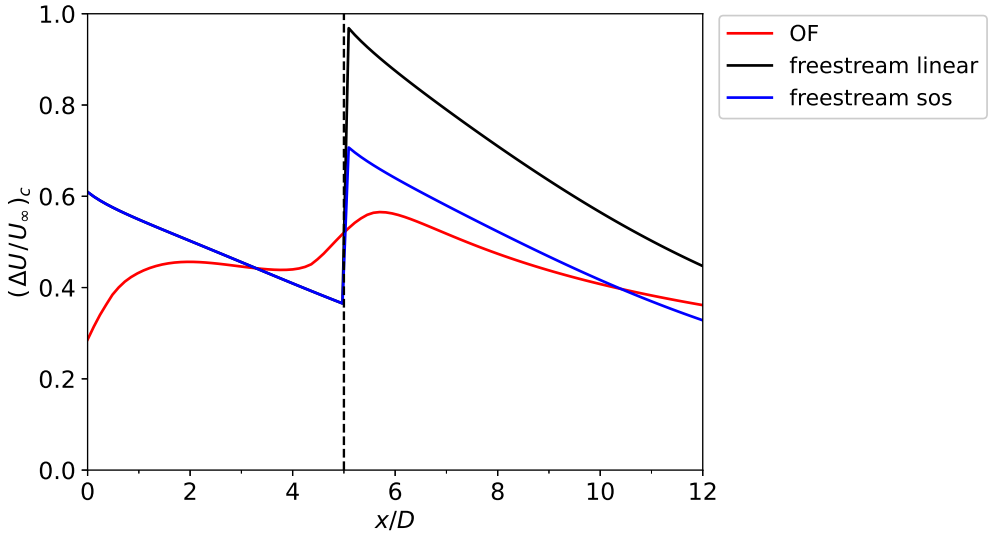


Figure 4.10: Centerline velocity deficit profiles illustrating the effect of the combination models (black and blue line) used together with the super-Gaussian wake model. The OF simulation is used as a reference (red line). The dashed line indicates the position of the second turbine (T1).

As seen in figure 4.10, there is a large difference between the combination models. Moreover, from the single-turbine validation, we know that the OF simulation overpredicts the velocity deficit for about $x/D \geq 4$ because of a too-low recovery rate. Therefore, the *freestream linear* result above is likely not accurate, because the predicted deficit is even larger than in the OF simulation, even at e.g. $x/D = 10$. Further, it can be observed from figure 4.10 that $(\Delta U/U_\infty)_c$ increases with about 0.61 at every turbine according to the *freestream linear* model. If a third turbine were to be included at $x/D = 10$, then this model would give a normalized velocity deficit larger than 1 (meaning that the wake velocity is in the opposite direction of the freestream) which is clearly unphysical. This shortcoming of the model has also been observed for horizontal-axis turbines [16]. The *freestream sos* model shows a better agreement with the OF simulation. However, one should be careful when making this comparison since the single-turbine results differ between the cases.

The combined wake of T0 and T1 is shown as cross-stream profiles in figure 4.11.

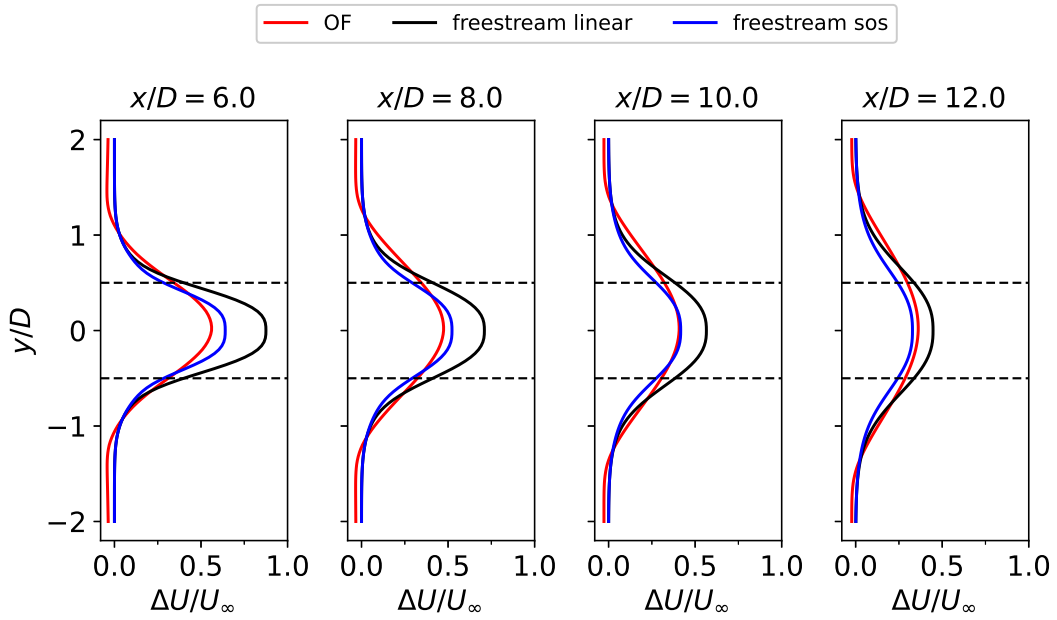


Figure 4.11: Velocity deficit profiles in the cross-stream direction, at several locations behind the second turbine (T1). Wake model results are compared for different combination models (black and blue lines). The OpenFOAM simulation is used as a reference (red lines). The dashed lines show the extent of the turbines.

Once again, the OF and *freestream sos* profiles are similar. However, the OF profile is slightly wider. This may be caused by the incoming turbulence intensity to T1, which is larger in the OF simulation (10.0 %, as mentioned in section 4.3.1.1) compared to the wake model (9.1 %).

Finally, the difference between the combination models are quantified in table 4.3 by considering the inflow velocity $U_{in,2}$ and generated power P_2 of a hypothetical third turbine T2 located at $x/D = 10$. The reference values $U_{in,2,ref}$ and $P_{2,ref}$ are obtained from the OpenFOAM simulation. Note that $P_2/P_{2,ref} = U^3_{in,2}/U^3_{in,2,ref}$ since the power coefficient used in the wake model calculations are the same as that given by CFD.

Table 4.3: Inflow velocity and power of a hypothetical third turbine T2 predicted by the wake model for different combination models. The OpenFOAM simulation is used as a reference.

Combination model	$U_{in,2}/U_{in,2,ref}$	$P_2/P_{2,ref}$
freestream linear	0.788	0.489
freestream sos	1.005	1.015

Table 4.3 shows that the *freestream sos* model predicts almost the same power as

the OpenFOAM simulation. Regarding the *freestream linear* model, there is a clear difference in the predicted velocity compared to the reference, which implies a very large difference in predicted power since it is proportional to the inflow velocity cubed.

4.3.2 Turbine array pair3

In this turbine array layout, the position of the second turbine (T1) is rotated 3° about the center of the first turbine (T0) to see how a change in wind direction affects the results from the previous section. To get a better understanding of the wake behind T1, cross-stream profiles are shown in figure 4.12 for the two combination models. Because of time limitations, a CFD simulation is not performed on this case.

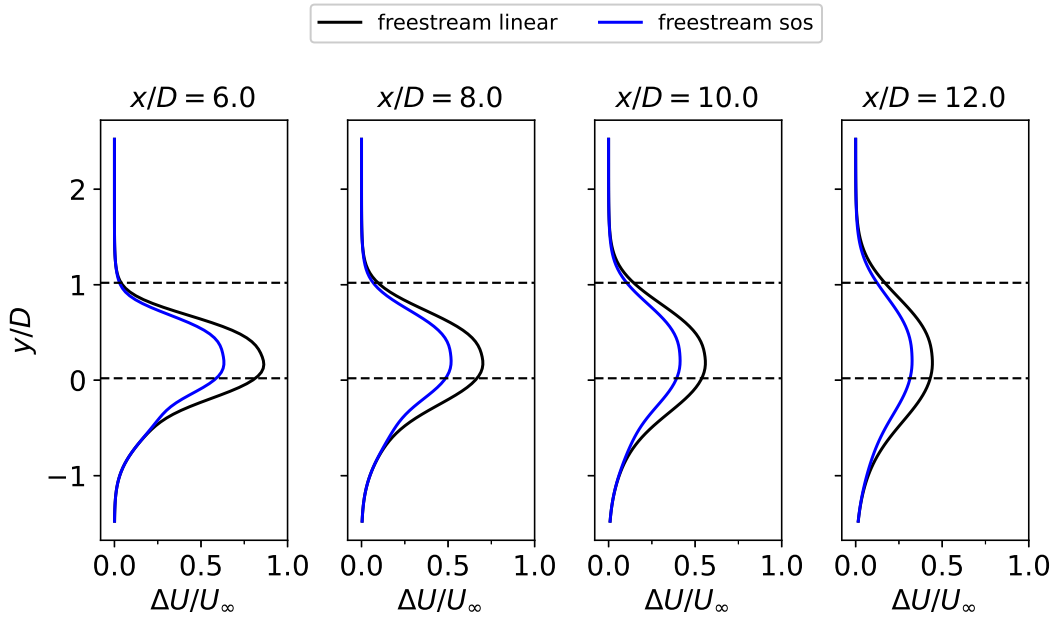


Figure 4.12: Velocity deficit profiles in the cross-stream direction, at several locations behind T1. Wake model results are compared for different combination models. The dashed lines show the extent of a hypothetical turbine T2 located at $x/D = 10$ and $y/D \approx 0.52$.

To obtain the inflow velocity and power of a hypothetical turbine T2, the velocity is sampled at $x/D = 10$ between the dashed lines shown in figure 4.12. Table 4.4 shows a comparison of the results between the combination models.

Table 4.4: Inflow velocity and power of a hypothetical third turbine T2 estimated by the *freestream linear* combination model. The *freestream sos* model is taken as the reference.

Combination model	$U_{in,2}/U_{in,2,ref}$	$P_2/P_{2,ref}$
freestream linear	0.839	0.590

From table 4.3 in the pair0 case it can be inferred that the difference in the predicted $U_{in,2}$ is about 22 % between *freestream linear* and *freestream sos*. In the *pair3* case, this difference is about 16 % (see table 4.4). The smaller difference between the combination models in the latter case is expected, since the more the layout is rotated, the less T0's wake affects T2. As a consequence, for an increased rotation angle, T2 will mostly be influenced by T1 (or by no turbine at all if the angle is large enough) which means that the effect of the combination models decays with the rotation angle.

5

Conclusion

This study set out to evaluate how well the super-Gaussian wake model by Ouro & Lazennec [6] predicts the velocity field and turbine powers in a vertical-axis wind turbine farm. As a first step, the wake model is implemented in FLORIS [11] which is an open-source wake model framework. This work thus extends the capability of FLORIS to vertical-axis wind turbines, since only horizontal axis-turbines were supported previously.

In a multi-turbine layout, a combination model is needed to combine the individual wakes given by the super-Gaussian model. The first major finding is that the wake model results depend heavily on the choice of combination model. This can be seen in the *pair0* layout, which consists of two turbines, with diameter D, spaced $5D$ apart in the streamwise direction. In this case, the difference between the *freestream linear* and *freestream sos* (sum-of-squares) combination model is 22 % in terms of inflow velocity to a hypothetical third turbine (located $5D$ behind the second turbine). The difference in predicted power for this hypothetical turbine is thus as large as 52 %.

To assess the performance of the wake model for different combination models, the results are compared to CFD simulations. Since the simulations are two-dimensional, the comparison is done in a horizontal plane at hub-height. The simulations are performed with OpenFOAM (an open-source CFD library) coupled with an actuator line model of the turbine blades. However, a limiting factor in the comparison is the accuracy of the CFD simulations. Specifically, the wake recovery rate is too low compared to a LES simulation by Abkar [9]. The $k - \varepsilon$ RANS model employed in this work suffers from turbulence decay, which means that the turbulence intensity decreases in the streamwise direction such that the value at the turbine location is significantly lower than the value prescribed at the inlet. In turn, a low incoming turbulence intensity to the turbine means a low recovery rate of the wake. Improvements could not be made by changing the inlet condition of the turbulent dissipation rate, but we learned that the simulations are sensitive to this parameter.

Notwithstanding the limitations above, the *freestream sos* combination model shows reasonable agreement with the CFD simulation on the *pair0* case. On the contrary, the wake velocity is likely under-estimated by the *freestream linear* model.

5.1 Future work

Since the wake model results are highly dependent on the combination model, and it is not clear which combination model to choose for a VAWT farm, there is a need for further research. It seems like a good strategy to validate the combination models against CFD simulations. In this way, it is possible to validate the CFD code on single-turbine experimental data (that covers both the near- and far-wake) and then use CFD simulations to extend the result to multi-turbine cases with possibly different wind directions and turbine tip-speeds ratios, which would otherwise be very time-consuming to do experimentally. In such a single-turbine validation, there is a need to explicitly validate the recovery rate in the CFD simulations. To this end, a three-dimensional LES + ALM setup is believed to be more accurate than the two-dimensional RANS + ALM setup used in this work. After completing the CFD validation, a natural continuation would be to repeat the study of Zong [16], where five combination models were compared on a three-turbine layout, but this time with vertical-axis turbines instead of horizontal-axis turbines. If a good combination model is found, then, and only then, can wake models be used to confidently predict the velocities in a VAWT farm and optimize the turbine layout to maximize power output.

Bibliography

[1] European Commission, "An EU strategy to harness the potential of offshore renewable energy for a climate neutral future," Brussels, Belgium, 52020DC0741, 2020. [Online]. Available: https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=COM:2020:741:FIN (visited on 2023-01-25).

[2] L. Kinning, "Sammanställning över planerad havsbaserad vindkraft i Sverige," Svensk Vindenergi, Stockholm, Sweden, 2022. [Online]. Available: https://svenskvindenergi.org/wp-content/uploads/2022/05/Sammanstallning-over-planerad-havsbaserad-vindkraft-2022-05-03-1.pdf (visited on 2023-01-25).

[3] S. Perez and M. Arvidsson, "Fortsatt hög elproduktion och elexport under 2021," Energimyndigheten, 2022. [Online]. Available: https://www.energimyndigheten.se/nyhetsarkiv/2022/fortsatt-hog-elproduktion-och-elexport-under-2021/ (visited on 2023-01-25).

[4] J. O. Dabiri, "Potential order-of-magnitude enhancement of wind farm power density via counter-rotating vertical-axis wind turbine arrays," *Journal of Renewable and Sustainable Energy*, vol. 3, no. 4, p. 043 104, 2011. DOI: 10.1063/1.3608170.

[5] J. T. Hansen, M. Mahak, and I. Tzanakis, "Numerical modelling and optimization of vertical axis wind turbine pairs: A scale up approach," *Renewable Energy*, vol. 171, pp. 1371–1381, 2021. DOI: 10.1016/j.renene.2021.03.001.

[6] P. Ouro and M. Lazennec, "Theoretical modelling of the three-dimensional wake of vertical axis turbines," *Flow*, vol. 1, p. E3, 2021. DOI: 10.1017/flo.2021.4.

[7] P. Bachant, A. Goude, and M. Wosnik, "Actuator line modeling of vertical-axis turbines," *arXiv*, 2016. DOI: 10.48550/ARXIV.1605.01449.

[8] V. Mendoza, P. Bachant, C. Ferreira, and A. Goude, "Near-wake flow simulation of a vertical axis turbine using an actuator line model," *Wind Energy*, vol. 22, no. 2, pp. 171–188, 2019. DOI: 10.1002/we.2277.

[9] M. Abkar and J. O. Dabiri, "Self-similarity and flow characteristics of vertical-axis wind turbine wakes: An LES study," *Journal of Turbulence*, vol. 18, no. 4, pp. 373–389, 2017. DOI: 10.1080/14685248.2017.1284327.

[10] NREL/floris#700 floris/simulation/wake_velocity/super_gaussian_vawt.py, [Contribution to source code]: G. Vallbo, [Online]. Available: https://github.com/NREL/floris/pull/700/files#diff-067d4f96c0b183af8d55e8726917f83a53d6fc8356b89cfad5dc84dad6ce9440.

[11] FLORIS, [Source code]: NREL, [Online]. Available: https://github.com/nrel/floris.

[12] NREL/floris#700 Add super-Gaussian velocity model for vertical-axis wind turbines, [Contribution to source code]: G. Vallbo, [Online]. Available: https://github.com/NREL/floris/pull/700.

[13] NREL/floris#701 Add infrastructure for vertical-axis wind turbines, [Contribution to source code]: G. Vallbo, [Online]. Available: https://github.com/NREL/floris/pull/701.

[14] NREL/floris#699 Add capability to sample and plot velocity deficit profiles, [Contribution to source code]: G. Vallbo, [Online]. Available: https://github.com/NREL/floris/pull/699.

[15] vallbog/floris/thesis, commit 8b7b0ad, [Source code]: G. Vallbo, [Online]. Available: https://github.com/vallbog/floris/tree/thesis.

[16] H. Zong and F. Porté-Agel, "A momentum-conserving wake superposition method for wind farm power prediction," *Journal of Fluid Mechanics*, vol. 889, p. A8, 2020. DOI: 10.1017/jfm.2020.77.

[17] M. Huang, "Wake and wind farm aerodynamics of vertical axis wind turbines," Phd thesis, Delft University of Technology, Delft, The Netherlands, 2023. DOI: 10.4233/uuid:14619578-e44f-45bb-a213-a9d179a54264.

[18] A. Bianchini, F. Balduzzi, P. Bachant, G. Ferrara, and L. Ferrari, "Effectiveness of two-dimensional CFD simulations for Darrieus VAWTs: A combined numerical and experimental assessment," *Energy Conversion and Management*, vol. 136, pp. 318–328, 2017. DOI: 10.1016/j.enconman.2017.01.026.

[19] turbinesFoam, [Source code]: P. Bachant, [Online]. Available: https://github.com/turbinesFoam/turbinesFoam.

[20] turbinesFoam#359 Add OpenFOAM v2212 support, [Source code]: P. Bachant, [Online]. Available: https://github.com/turbinesFoam/turbinesFoam/pull/359.

[21] NTNU-HAWT-turbinesFoam, [Source code]: P. Bachant, [Online]. Available: https://github.com/petebachant/NTNU-HAWT-turbinesFoam.

[22] J. N. Sørensen and W. Z. Shen, "Numerical modeling of wind turbine wakes," *Journal of Fluids Engineering*, vol. 124, no. 2, pp. 393–399, 2002. DOI: 10.1115/1.1471361.

[23] M. Churchfield, S. Schreck, L. A. Martínez-Tossas, C. Meneveau, and P. R. Spalart, "An advanced actuator line method for wind energy applications and beyond," in *Proceedings of the American Institute of Aeronautics and Astronautics SciTech 2017*, (Grapevine, Texas, USA), 2017. [Online]. Available: https://www.nrel.gov/docs/fy17osti/67611.pdf (visited on 2023-05-10).

[24] S. T. Fredriksson, G. Broström, B. Bergqvist, J. Lennblad, and H. Nilsson, "Modelling Deep Green tidal power plant using large eddy simulations and the actuator line method," *Renewable Energy*, vol. 179, pp. 1140–1155, 2021. DOI: 10.1016/j.renene.2021.07.058.

[25] vallbog/turbinesFoam/forceProjection2D_2112_epsMethod, commit 64697ba, [Contribution to source code]: G. Vallbo, [Online]. Available: https://github.com/vallbog/turbinesFoam/tree/forceProjection2D_2112_epsMethod.

[26] M. Abkar, "Theoretical modeling of vertical-axis wind turbine wakes," *Energies*, vol. 12, no. 1, 2019. DOI: 10.3390/en12010010.

[27] E. Möllerström, F. Ottermo, A. Goude, S. Eriksson, J. Hylander, and H. Bernhoff, "Turbulence influence on wind energy extraction for a medium size vertical axis wind turbine," *Wind Energy*, vol. 19, no. 11, pp. 1963–1973, 2016. DOI: 10.1002/we.1962.

[28] F. M. White, *Fluid mechanics*, 8th ed. New york, NY, USA: McGraw-Hill Education, 2016.

[29] F. Balduzzi, A. Bianchini, R. Maleci, G. Ferrara, and L. Ferrari, "Critical issues in the CFD simulation of Darrieus wind turbines," *Renewable Energy*, vol. 85, pp. 419–435, 2016. DOI: 10.1016/j.renene.2015.06.048.

[30] S. Shamsoddin and F. Porté-Agel, "A large-eddy simulation study of vertical axis wind turbine wakes in the atmospheric boundary layer," *Energies*, vol. 9, no. 5, 2016. DOI: 10.3390/en9050366.

[31] H. K. Versteeg and W. Malalasekera, *An introduction to computational fluid dynamics: The Finite volume method*, 2nd ed. Harlow, UK: Pearson, 2007.

[32] C. Rumsey, "The Menter shear stress transport turbulence model," in *Turbulence modeling resource*, Hampton, VA, USA: NASA Langley Research Center, 2023. [Online]. Available: https://turbmodels.larc.nasa.gov/sst.html (visited on 2023-10-09).

[33] L. Davidson, *Fluid mechanics, turbulent flow and turbulence modeling*. Gothenburg, Sweden: Chalmers University of Technology, 2023. [Online]. Available: https://www.tfd.chalmers.se/~lada/postscript_files/solids-and-fluids_turbulent-flow_turbulence-modelling.pdf.

[34] H. Nilsson, "Chapter_3_presented: Turbulence modeling," MTF072 lecture notes, Chalmers University of Technology, Gothenburg, Sweden, 2020.

[35] R. E. Sheldahl and P. C. Klimas, "Aerodynamic characteristics of seven symmetrical airfoil sections through 180-degree angle of attack for use in aerodynamic analysis of vertical axis wind turbines," Sandia National Laboratories, Albuquerque, NM, USA, SAND80-2114, 1981. DOI: 10.2172/6548367.

[36] E. Dyachuk, A. Goude, and H. Bernhoff, "Dynamic stall modeling for the conditions of vertical axis wind turbines," *AIAA Journal*, vol. 52, no. 1, 2014. DOI: 10.2514/1.J052633.

[37] C. Greenshields, "Numerical schemes," in *OpenFOAM v11 user guide*, London, UK: The OpenFOAM Foundation, 2023. [Online]. Available: https://doc.cfd.direct/openfoam/user-guide-v11/fvschemes (visited on 2023-10-09).

[38] D. Monni, "Numerical simulating and analytical modelling the wake behind vertical axis wind turbines: A verification analysis," Master thesis, Politecnico di Torino, Turin, Italy, 2021. [Online]. Available: https://webthesis.biblio.polito.it/18899/.

[39] M. Kinzel, D. B. Araya, and J. O. Dabiri, "Turbulence in vertical axis wind turbine canopies," *Physics of Fluids*, vol. 27, no. 11, p. 115 102, 2015. DOI: 10.1063/1.4935111.

[40] UNH-RVAT-turbinesFoam, commit 9a034c9, [Source code]: P. Bachant, [Online]. Available: https://github.com/petebachant/UNH-RVAT-turbinesFoam.

[41] Y.-T. Wu and F. Porté-Agel, "Atmospheric turbulence effects on wind-turbine wakes: An LES study," *Energies*, vol. 5, no. 12, pp. 5340–5362, 2012. DOI: 10.3390/en5125340.

[42] S. Shamsoddin and F. Porté-Agel, "Large eddy simulation of vertical axis wind turbine wakes," *Energies*, vol. 7, no. 2, pp. 890–912, 2014. DOI: 10.3390/en7020890.

A Definition of additional wake model variables

For completeness, the maximum velocity deficit $C(x)$ and the initial characteristic wake widths ε_y and ε_z are defined below, according to Ouro & Lazennec [6]. Note that all of these variables are dimensionless. The maximum velocity deficit is given as

$$C(x) = 2^{\eta-1} - \sqrt{2^{2\eta-2} - \frac{C_T n_y n_z}{8\widetilde{\sigma}_y^{2/n_y}\widetilde{\sigma}_z^{2/n_z}\Gamma(1/n_y)\Gamma(1/n_z)}}$$

were C_T is the turbine thrust coefficient, $\eta = 1/n_y + 1/n_z$ and Γ is the gamma-function.

Further, it is assumed that $\varepsilon_y = \varepsilon_z = \varepsilon$, where ε is only a function of C_T and the model parameters:

$$\varepsilon = \left(\frac{\beta n_{y,0} n_{z,0}}{2^{2\eta_0+2}\Gamma(1/n_{y,0})\Gamma(1/n_{z,0})}\right)^{\frac{1}{2\eta_0}}$$

where

$$n_{y,0} = n_y(x - x_c = 0) = a_y + c_y, \quad n_{z,0} = n_z(x - x_c = 0) = a_z + c_z$$

$$\eta_0 = \frac{1}{n_{y,0}} + \frac{1}{n_{z,0}}, \quad \beta = \frac{1}{2}\frac{1+\sqrt{1-C_T}}{\sqrt{1-C_T}}$$

To have the initial characteristic wake widths the same in the cross-stream and vertical direction ($\varepsilon_y = \varepsilon_z$) might seem strange at first, but remember that these quantities are dimensionless. From the *dimensional* characteristic wake widths σ_y and σ_z, it is clear that the initial width is proportional to the turbine diameter D in y and to the blade height H in z as expected:

$$\sigma_y = D\widetilde{\sigma}_y = k_y^*(x - x_c) + D\varepsilon, \quad \sigma_z = H\widetilde{\sigma}_z = k_z^*(x - x_c) + H\varepsilon$$

B How to sample and plot velocity deficit profiles in FLORIS: A minimal code example

This work adds the ability to sample and plot velocity deficit profiles in FLORIS. The functionality is used to create all the figures of *cross-stream profiles* in this report.

Listing 2 shows an example of how to recreate the basics of figure 4.3 (only including the FLORIS results for simplicity). Note that the `FlorisInterface` class is the main way to interact with FLORIS. In this case, we instantiate the class by providing a path to the input file where wind speed, wind direction, turbine parameters, velocity model, etc. is provided. Then, the added method `sample_velocity_deficit_profiles` and the added class `VelocityProfilesFigure` is used to create the figure.

```
import matplotlib.pyplot as plt
import numpy as np
from floris.tools import FlorisInterface
from floris.tools.visualization import VelocityProfilesFigure

fi = FlorisInterface('inputs/super_gaussian_vawt.yaml')

D = 26.0 # [m]. Turbine diameter
# For each specified downstream distance from a starting point, sample
# a velocity profile along a line in the cross-stream direction (y).
# The starting point (x_inertial_start, y_inertial_start) is set to
# the location of the turbine hub which is (0.0, 0.0)
downstream_dists = np.array([3, 6, 9, 12]) * D
profiles = fi.sample_velocity_deficit_profiles(
    direction='y',
    downstream_dists=downstream_dists,
    profile_range=np.array([-2, 2]) * D,
    x_inertial_start=0.0,
    y_inertial_start=0.0,
)
fig = VelocityProfilesFigure(
    downstream_dists_D=downstream_dists / D,
    layout=['y'],
)
fig.add_profiles(profiles)
fig.set_xlim([-0.08, 0.52])

plt.show()
```

Listing 2: Example of how to sample and plot velocity deficit profiles in FLORIS.

C Derivation of a 2d Gaussian function for force projection

The integral in (3.5) is performed in polar coordinates $(r,\, \theta)$ where $dA = rd\theta dr$:

$$\int_{\mathbb{R}^2} \eta_{\varepsilon,2d}(\boldsymbol{x})\, dA = \int_0^{\infty} \int_0^{2\pi} Ce^{-\frac{r^2}{\varepsilon^2}} r d\theta dr = 2\pi C \int_0^K e^{-\frac{r^2}{\varepsilon^2}} r\, dr =$$

$$\left\{ s = r^2,\ \frac{ds}{dr} = 2r,\ s(0) = 0,\ s(K) = K^2 \right\} = \pi C \int_0^{K^2} e^{-\frac{s}{\varepsilon^2}}\, ds = \pi C \left[\frac{e^{-\frac{s}{\varepsilon^2}}}{-\frac{1}{\varepsilon^2}} \right]_0^{K^2} =$$

$$-\pi\varepsilon^2 C \left(\frac{1}{e^{\frac{K^2}{\varepsilon^2}}} - 1 \right) \underset{K\to\infty}{\longrightarrow} \pi\varepsilon^2 C$$

The integral should be equal to 1, which means that $C = \frac{1}{\pi\varepsilon^2}$ and thus:

$$\eta_{\varepsilon,2d}(\boldsymbol{x}) = \frac{1}{\pi\varepsilon^2} e^{-\frac{|\boldsymbol{x}|^2}{\varepsilon^2}}$$

D Sensitivity of turbulence model inlet conditions: Cross-stream profiles

Additional results from the $\varepsilon_{turb,in}$ sensitivity analysis are shown in the figures below. Let C_T be the average thrust coefficient during the last turbine revolution. The difference in C_T between $\varepsilon^*_{turb,in} \in [0.5, 0.1, 0.025]$ and $\varepsilon^*_{turb,in} = 1$ is less than 10 %, meaning that roughly the same blade forces are applied to the flow in all cases. However, the velocity deficit is spread out a lot more for lower $\varepsilon^*_{turb,in}$ (see figure D.1), indicating large values of momentum diffusion caused by a high turbulent viscosity ν_t.

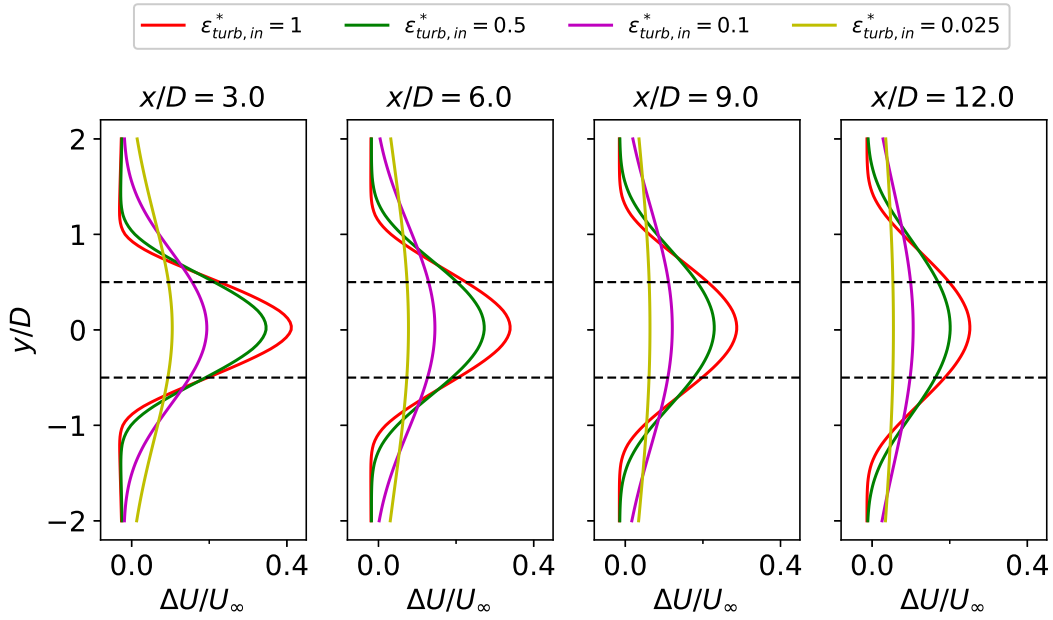


Figure D.1: Cross-stream profiles of velocity deficit for different inlet values $\varepsilon_{turb,in}$.

Similar to above, the diffusion of turbulent kinetic energy also increases dramatically for lower $\varepsilon^*_{turb,in}$ (see figure D.2). For example, we know from before that $\varepsilon^*_{turb,in} = 0.025$ gives $k/k_{in} \approx 0.90$ at $x/D = 0$ (see figure 4.7). However, for $|y/D| \leq 2D$, we see that k/k_{in} is clearly larger than 1 for all downstream distances in figure D.2. This must be caused by large diffusion of the turbulent kinetic energy generated by the turbine.

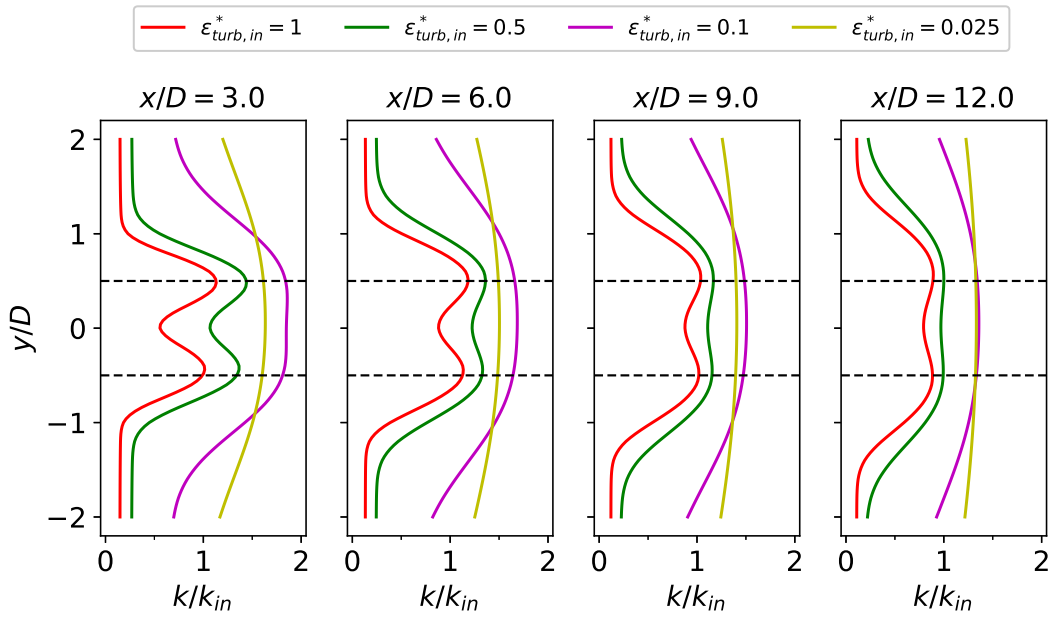


Figure D.2: Cross-stream profiles of turbulent kinetic energy for different inlet values $\varepsilon_{turb,in}$.

DEPARTMENT OF MECHANICS AND MARITIME SCIENCES
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden
www.chalmers.se

CHALMERS
UNIVERSITY OF TECHNOLOGY