

Article

A Steady-State Wind Farm Wake Model Implemented in OpenFAST

Antonio Cioffi , Claudia Muscari , Paolo Schito  and Alberto Zasso 

Department of Mechanical Engineering, Politecnico di Milano, 20156 Milano, Italy;
claudia.muscari@polimi.it (C.M.); paolo.schito@polimi.it (P.S.); alberto.zasso@polimi.it (A.Z.)

* Correspondence: antonio.cioffi@polimi.it; Tel.: +39-389-5690668

Received: 14 October 2020; Accepted: 17 November 2020; Published: 24 November 2020



Abstract: Wake models play a fundamental role in finding optimized solutions in wind farm control. In fact, they allow assessing how wakes develop and interact with each other with the agility required for real-time applications. In this paper, a Gaussian Wake Model (GWM) is implemented in the OpenFAST framework in a way such that its fidelity is increased with respect to previously implemented models, while enhancing its compatibility with control purposes. The OpenFAST tool is coupled with Floris, NREL's software based on the GWM, in order to simulate the wake effect on downstream machines (in the case where the downstream rotor is fully covered by the wake, only partially covered by the wake, of the wake is generated by the interaction of more than one turbine), while the rotor aerodynamics is calculated using the BEMT on the actual rotor flow field. We intend this work as a starting point for developing and testing open/closed-loop control logics that will work in real wind farms. To show the suitability of the implementation, the entire model is then compared to Floris.

Keywords: wind farm control; wind turbine; wake model; Floris; OpenFAST; Gaussian wake model

1. Introduction

Wind energy installations are continuously growing, and the available sites where consistent wind is blowing are limited; this typically leads to grouping wind turbines into clusters. However, in order to minimize the rental cost, the owners of the farm will place the turbines very close to one another, promoting in this way the interaction of wakes from upstream machines that negatively affect the power extraction and the loads acting on the turbines placed downstream. This is the reason why, before placing the turbines, an accurate study of the wind blowing in the site is needed to correctly choose the best layout of the farm based on how the wakes could develop.

When the wind farm is in operation, a control-oriented wake model (capable of running in real time if needed for control purposes) could be used to optimize both the power extraction and the fatigue loads acting on the machines for all the possible environmental conditions found at the site. These topics, as indicated in van Kuik et al. [1], are active important research topics in the wind energy field. Currently, many studies are dedicated also to the development of wind plant models having different levels of complexity and fidelity, based on the purposes for which they are used. Among these models, there are on the one side high fidelity wake models. They are particularly suited for model validation since they are able to grasp the physics behind the phenomenon represented. On the other side, there are engineering and analytical models that give a more rough estimation of the flow field or turbine performance, but are still valuable for the evaluation of turbine performances.

As of today, the state-of-the-art of high fidelity methods is the coupling between OpenFAST [2] (an aero-elastic wind turbine simulator) and OpenFOAM/SOWFA [3,4] (an open-source CFD solver with an ad hoc library able to reproduce the atmospheric boundary layer) using an Actuator Line Model (ALM) [5]. By using this tool, it is possible to estimate the turbines' load and behavior in the closest way to what happens

in reality. However, when dealing with control problems, we must renounce some of the complexity in order to save time and computational power. This is why low fidelity models are usually analytical ones.

An example of open-source control-oriented software is Floris [6], developed by NREL. It relies on the Bastankhah and Porté-Agel [7,8] GWM and is able to fully characterize the flow field developed in a wind farm, considering steady-state environmental conditions. To work, this tool requires only four parameters determined experimentally and some inputs from the user related mainly to the farm geometry and the free stream wind. There also exists a compromise. Some models are able to guarantee both good accuracy in the results and a fast execution of the simulations. Our intention with this work is to place ourselves in this area. Here, we can already find the Dynamic Wake Meandering (DWM) [9] model developed at DTU (Danmarks Tekniske Universitet) and the WindFarmSimulator (WFSim) developed at TU Delft. The first is derived from a simplified version of the Navier–Stokes equations and is able to reproduce the wake meandering phenomenon, which has a big impact as regards the power extraction and loads of the machines. The second one is based on the 2D Navier–Stokes equations and has been widely used, together with Floris, to investigate new control algorithms. These studies supported the concept of secondary steering and showed how wake steering methods could be used for multi-objective optimization in the wind farm context [10,11].

In this paper, a low fidelity model based on the GWM and Floris software is presented; it is more advanced, however, since the GWM will be embedded and working in the OpenFAST framework coupled to the BEMT aerodynamic model of the turbine. The main difference with Floris is the capability of this new tool to model the wake generated by the machines taking into account the C_T and AIF developed on the whole plane of the rotor and calculated at runtime from the aero-elastic solver OpenFAST, not only at a single point (at hub-height) by means of pre-tabulated values like the $C_P - TSR$ or the $C_T - TSR$ table (like Floris does). This allows for a more accurate description of the wake. The proposed tool allows also for a characterization of the loads acting on the machines when they work in the complete or partial wake condition under the assumption of steady-state environmental conditions. In addition to this, the model can be used to develop and test new control logics at the wind farm level in order to maximize the power extraction under the hypothesis of the steady-state flow condition, but retaining the dynamics of the turbines (thanks to OpenFAST). It is also possible to investigate both the loads and the power extraction when the turbines are working in a yawed condition. Indeed, this new tool is particularly suited for research on wind farm control logics using wake steering methods, like the one proposed by van Dijk et al. [10] and Martinez et al. [11], axial-induction based methods, like the one presented by van Wingerden et al. [12], or a combination of them like in the work of Bossanyi et al. [13].

The paper is organized as follows. First, we give an overview of the GWM and the analytical changes that have been made to it. Then, the next section is dedicated to the implementation of the new GMW in the OpenFAST framework. Lastly, we present of a simulation carried out with the new developed tool and a discussion about the results and the future work.

2. Theory

2.1. GWM Overview

The GWM model proposed by Basthakhah and Porté-Agel [7,8] relies on the hypothesis of the self-similarity of the wakes, meaning that the wake velocity deficit profile shows always the same characteristics if properly normalized. The entire model, however, is derived from a budget study of the Navier–Stokes equations in their RANS formulation.

For the profile of the wake deficit, a Gaussian shape is proposed. This hypothesis is valid, as demonstrated by the authors, for a relative yaw angle of the turbine rotor, with respect to the free-stream wind direction, up to 30 degrees; after this limit value, the wake profile starts to differ significantly from the Gaussian shape. In addition to a complete characterization of the wake deficit, the authors also proposed an analytical description of the wake deflection when the turbine is operating in the yawed condition. Specifically, the analytical formulation for the wake's deficit and deflection in the far wake zone is:

$$\frac{\bar{u}(x, y, z)}{u_\infty} = 1 - C(x) e^{-\frac{(y-\delta)^2}{2\sigma_y^2}} e^{-\frac{(z-z_h)^2}{2\sigma_z^2}} \quad (1)$$

$$\frac{\theta(x, y, z)}{\theta_m} = e^{-\frac{(y-\delta+\sigma_y)^2}{2\sigma_y^2}} e^{-\frac{(z-z_h)^2}{2\sigma_z^2}} \quad \text{and} \quad \theta_m = \frac{\theta_c}{e^{-0.5}} \quad (2)$$

where \bar{u} represents a time averaged velocity in a downstream location with respect to the considered turbine, $C(x)$ represents the normalized velocity deficit at the wake center in various downstream positions, δ represents the wake center deflection, z_h represents the hub height of the considered wind turbine, σ_y and σ_z represent the wake width in the y and z directions, u_∞ represents the undisturbed fluid velocity, and θ_c represents the wake skew angle of the wake center as a function of the downstream distance.

As regards the wake width, σ_y and σ_z , a linear formulation dependent on the downstream distance is proposed:

$$\frac{\sigma_z}{d} = k_z \frac{x - x_0}{d} + \frac{\sigma_{z0}}{d} \quad (3)$$

$$\frac{\sigma_y}{d} = k_y \frac{x - x_0}{d} + \frac{\sigma_{y0}}{d} \quad \text{and} \quad \frac{\sigma_{y0}}{d} = \frac{\sigma_{z0}}{d} \cos \gamma \quad (4)$$

$$k_y = k_z = k_a I + k_b \quad \text{and} \quad k_a = 0.38371, k_b = 0.003678 \quad (5)$$

where d represents the turbine's diameter, x_0 the downstream distance at which there is the onset of the far wake zone, σ_{y0} and σ_{z0} represent the wake width at the onset of the far wake zone, k_z and k_y are two constants hypothesized as linearly dependent from the wind turbulence intensity I , k_z and k_y are two constants determined experimentally, and γ is the yaw angle of the turbine relative to the free-stream wind direction. Then, applying two conditions directly derived from the budgeting operation carried out on the RANS equations, in the stream-wise and span-wise direction on the expressions in Equations (1) and (2) (which is also integrated in the stream-wise direction), it is possible to state that:

$$C(x) = 1 - \sqrt{1 - \frac{\sigma_{y0}\sigma_{z0}M_0}{\sigma_y\sigma_z}} \quad (6)$$

$$\delta = \delta_0 + \frac{\theta_{c0}E_0}{5.2} \sqrt{\frac{\sigma_{y0}\sigma_{z0}}{k_yk_z}M_0} \times \ln \frac{(1.6 + \sqrt{M_0})(1.6\sqrt{\frac{\sigma_y\sigma_z}{\sigma_{y0}\sigma_{z0}}} - \sqrt{M_0})}{(1.6 - \sqrt{M_0})(1.6\sqrt{\frac{\sigma_y\sigma_z}{\sigma_{y0}\sigma_{z0}}} + \sqrt{M_0})} \quad (7)$$

where $M_0 = C_0(2 - C_0)$, $E_0 = C_0^2 - 3e^{\frac{1}{12}}C_0 + 3e^{\frac{1}{3}}$, C_0 is the wake deficit speed at the onset of the far wake region, δ_0 is the wake deflection with respect to the turbine's rotor axis at the onset of the far wake region, and θ_{c0} is the wake skew angle of the wake center at the onset of the far wake region. Now, to close the model, it is necessary to characterize all the parameters referring to the onset of the far wake region, which are: C_0 , x_0 , δ_0 , σ_{y0} , σ_{z0} , and θ_{c0} .

2.2. Determining the Parameters in the Far Wake Region

A relationship between the C_T and the a (AIF) of the turbine is needed; the authors of the GWM proposed a simplified formulation:

$$C_T \simeq 4a(1 - a \cos \gamma) \quad \text{and} \quad a \simeq \frac{1}{2 \cos \gamma} (1 - \sqrt{1 - C_T \cos \gamma}) \quad (8)$$

This analytical formulation, as will be seen in Section 2.4, is fundamental in the formulation of the modified GWM model to be implemented in OpenFAST, since now, the aero-elastic code will directly provide the C_T and AIF. In the Floris software, to run the model, a $C_T - TSR$ and a $C_p - TSR$ table are required as input parameters. Thanks to them, the AIFs of the simulated turbines are derived

by means of Equation (8), using the input wind speed to define the TSRs at which the machines are operating ($TSR \rightarrow C_T \rightarrow AIF$).

Continuing with the Bastankhah and Porté-Agel [7,8] model, applying the definition of AIF, it can be stated that:

$$\frac{u_R}{u_\infty} = \frac{C_T \cos \gamma}{2(1 - \sqrt{1 - C_T \cos \gamma})} \quad (9)$$

where u_R is the wind speed at the turbine's rotor plane.

Applying the Bernoulli equation between an upstream (free-stream wind) section and the rotor section of the turbine and between the rotor section and a downstream section, it is possible to find exactly that:

$$\frac{u_0}{u_\infty} = \sqrt{1 - C_T} \quad (10)$$

From this considerations, the theoretical value of the deficit at the rotor plane section C_0 can be expressed as:

$$C_0 = 1 - \frac{u_0}{u_\infty} = 1 - \sqrt{1 - C_T} \quad (11)$$

Now, assuming that the "potential core" of the wake (the near wake zone) has a constant velocity equal to u_0 , it is possible to find, applying the budgeted RANS conditions at section $x = x_0$ (thus equating the expression of the potential core and the Gaussian shape at the section in which there is the onset of the far wake region), that:

$$\frac{\sigma_{z0}}{d} = \frac{1}{2} \sqrt{\frac{u_R}{u_\infty + u_0}} \quad (12)$$

and applying Equations (9) and (10):

$$\frac{\sigma_{z0}}{d} = \sqrt{\frac{1 + \sqrt{1 - C_T \cos \gamma}}{8(1 + \sqrt{1 - C_T})}} \simeq \sqrt{\frac{1}{8}} \quad (13)$$

Passing to the wake deflection and using Burton's approximation [14] to describe the flow skew angle with respect to the rotor's axis χ :

$$\chi = \theta + \gamma = (0.6a + 1)\gamma \quad (14)$$

and by substituting the AIF with Equation (8):

$$\theta \simeq \frac{0.3\gamma}{\cos \gamma} (1 - \sqrt{1 - C_T \cos \gamma}) \quad (15)$$

The θ angle should be used to estimate the flow angle at the rotor disk, but it also proves to be a good prediction for the flow skew angle up to a few rotors' diameters downstream; this is why Equation (15) is used to define the parameter θ_{c0} (the flow skew angle at the onset of the far wake region, where $x = x_0$). From this consideration, it is easy to see how the wake deflection δ_0 at $x = x_0$ can be expressed as:

$$\delta_0 = x_0 \tan \theta_{c0} \simeq x_0 \theta_{c0} \quad (16)$$

The last expression to be determined is x_0 , the downstream distance at which there is the onset of the far wake region; to determine this unknown, the authors adopted a model from Lee and Chu [15]. The model suggests a proportionality between the change in width of the shear layer σ_{y0} to the velocity difference between the potential core of the wake and the unperturbed surroundings. They also

recognized a proportionality of σ_{y0} with the incoming turbulence I that enhances flow entrainment and the growth of the shear layer. Therefore, by integrating the following expression:

$$\frac{1}{u_\infty} \frac{ds}{dt} = \frac{u_s}{u_\infty} \frac{ds}{dx} = \alpha I + \beta \frac{u_e}{u_\infty} \quad (17)$$

$$u_s = 0.5(u_\infty + u_0) \quad \text{and} \quad u_e = 0.5(u_\infty - u_0) \quad (18)$$

and applying the proper substitutions, they were able to define x_0 as:

$$\frac{x_0}{d} = \frac{\cos \gamma (1 + \sqrt{1 - C_T})}{\sqrt{2} [4\alpha I + 2\beta (1 - \sqrt{1 - C_T})]} \quad (19)$$

2.3. Additional Considerations

The GWM model described above is used to reproduce the wake generated by a single turbine; in the case of a wind farm, it has to be considered that an interaction between wakes may happen.

In the simple case of three turbines aligned in the wind direction and positioned one downstream of the other (like in the simulation proposed in this paper), for example, it is easy to see that it is not only the wake of the second machine having an effect on the wind velocity felt by the last one, but also the wake developed from the first. This is why a method that accounts for this phenomenon and the added turbulence generated by the wakes is necessary. In this subsection, the methods used are exactly the ones proposed in the open-source Floris software developed by NREL. The correction used for the wind velocity was developed by Katic et al. [16] and proposes a superimposition of the local wakes to estimate the wake deficit on the $n + 1$ downstream turbine:

$$\Delta u_{n+1} = \sqrt{\sum_{i=1}^n \Delta u_i^2} \quad (20)$$

Crespo et al. [17] proposed a correction for the added turbulence due to the wake generation and mixing with the free-stream velocity:

$$I_+ = C_I A_{ol,\%} a_i^{e_1} I_0^{e_2} \left(\frac{x}{D_i} \right)^{e_3} \quad (21)$$

where I_0 is the ambient turbulence intensity referred to the free stream condition and C_I and e_1 , e_2 , and e_3 are parameters to be determined experimentally that in Floris were set to $C_I = 0.8$, $e_1 = 0.73$, $e_2 = 0.1$, and $e_3 = -0.275$. This last relation was derived experimentally, while the first one is linked to dynamic pressure and to the thrust force exerted by the upstream turbines. The $A_{ol,\%}$ term appearing in the I_+ equation represents the ratio of overlapping area between the rotor diameter of a turbine and the wake area generated by upstream machines, while D_i , a_i and I_0 represent respectively the rotor diameter, the Axial Induction Factor (AIF), and the incoming turbulence intensity on the i -th considered machine. Furthermore, for the turbulence intensity I , due to the wake's superposition, a correction by Niayifar et al. [18] was implemented:

$$I = \sqrt{\sum_{i=1}^n I_{+,i}^2 + I_0^2} \quad (22)$$

In Floris, when evaluating the power extracted from a turbine in a yawed configuration, another correction is used; considering that the C_P (power coefficient) of the machine is proportional to the cube of the incoming undisturbed flow speed, yawing the turbine directly reduces the useful wind component from which power is extracted by the cosine of the misalignment angle:

$$C_{P,yaw} = C_P \cos^{p_p} \gamma \quad (23)$$

The same is valid for C_T :

$$C_{T,yaw} = C_T \cos^{p_t} \gamma \quad (24)$$

Both p_p and p_t need to be experimentally derived. These corrections aim at adjusting the characteristic coefficients of the turbine when working in yawed conditions in a simple and effective way. However, with the implementation of the GWM in OpenFAST, the p_p and p_t coefficients will not be needed anymore since the aero-elastic code is perfectly capable of evaluating the $C_{P,yaw}$ and $C_{T,yaw}$ directly, without any need for corrections.

2.4. Changes to the GWM

As specified in Section 2.2, to implement the GWM and the necessary corrections due to the wakes' interaction in the OpenFAST software, some changes were necessary to the GWM proposed by Bastankhah and Porté-Agel [7,8]. The major change is that both the C_T and AIF can be retrieved directly from the program as outputs of the aerodynamics calculations of the BEMT, running on each simulated turbine. We can treat OpenFAST as a black box that, given certain inputs as the wind speed acting on the rotor plane, the lift and drag coefficients of the blades, etc., outputs certain quantities like the C_T and AIF, needed to run the GWM model. Therefore, the analytical formulation in Equation (8) and all the mathematical steps deriving from it should be discarded when using OpenFAST; this means that Equation (9) has to be changed into:

$$\frac{u_R}{u_\infty} = 1 - a \quad (25)$$

since OpenFAST directly outputs the a value and Equation (13) has to be rewritten as:

$$\frac{\sigma_{z0}}{d} = \frac{1}{2} \sqrt{\frac{1-a}{1+\sqrt{1-C_T}}} \quad (26)$$

while Equation (15) is changed into:

$$\theta = 0.6a\gamma \quad (27)$$

and finally, Equation (19) is written as:

$$\frac{x_0}{d} = \frac{\cos \gamma \sqrt{(1-a)(1+\sqrt{1-C_T})}}{4\alpha I + 2\beta(1-\sqrt{1-C_T})} \quad (28)$$

After applying these corrections to the GWM, the model is finally ready to be embedded in the OpenFAST framework.

3. Implementing the GWM in OpenFAST

OpenFAST allows customizing some functions based on specific needs: one of them is the wind model. It is possible to write in the InflowWind Module of OpenFAST, responsible for modeling the wind acting on the machine, a user-defined routine in a specific file called `IfW_UserWind.f90`, in Fortran language. OpenFAST, like the above-mentioned routine, is completely written in Fortran due to its speed in executing calculations, except for its API, which is written in C++. The API enables the software to be coupled with OpenFOAM and SOWFA to simulate a turbine, or a complete wind farm immersed in the ABL, or to simply simulate more than one machine at the same time (in a serial fashion, not parallel). This is the reason why this API will be used when performing a complete simulation of a wind farm with the user-specified GWM model. The implemented function basically executes this task: given as input an array of positions near the rotor of each turbine (needed to solve

the aerodynamics (BEMT) of the machine), the routine returns as output a matrix of values containing the wind speed in the three directions x , y , and z for the selected positions.

A brief description of the implemented routine is reported in Appendix A.

In the main function, in addition to the GWM implementation, it was necessary to describe the free-stream wind acting on the first upstream turbine; to do so, the model chosen was the logarithmic profile of wind described in the EUROCODE [19]. The wind speed and the ambient turbulence intensity acting at a given height z are given by the following equations:

$$u_{\infty}(z) = V_b k_r \ln\left(\frac{z}{z_0}\right), \quad z_{\min} \leq z \leq 200 \text{ m}/300 \text{ m} \quad (29)$$

$$u_{\infty}(z) = u_{\infty}(z_{\min}), \quad z < z_{\min} \quad (30)$$

$$k_r = 0.19 \left(\frac{z_0}{0.05}\right)^{0.07} \quad (31)$$

$$I_0 = \frac{1}{\ln\left(\frac{z}{z_0}\right)} \quad (32)$$

where z_0 is the surface roughness, V_b is the basic wind velocity, $z_{\min} \in [1 - 10]$ m is the minimum height at which the hypothesis of logarithmic profile is verified, and k_r is the terrain factor. In Figure 1, a schematic representation of how the main routine is intended to work, when coupled with the OpenFAST framework, is presented. For a generic time step, starting from the most upstream turbine simulated by the first instance of OpenFAST, the user-specified InflowWind routine evaluates the free-stream wind speed acting on the positions required by OpenFAST; then, the array containing the wind velocities is passed to the AeroDyn Module to solve the aerodynamics of the machine (using the BEMT model). After this passage, AeroDyn produces as output the $C_{T,i}$ value for the current turbine together with the AIF values at specific blades' section $a_{local,i}(r)$ specified in the input file of AeroDyn (for a maximum of nine specific sections per blade). Then, the current OpenFAST instance (that is simulating the first turbine) continues doing its calculations for the specific time step; once finished, the second instance of OpenFAST, simulating the first downstream turbine, is called to perform all its tasks. This can be performed because the hypothesis is that the turbine is not affecting the upstream wind field, but only the wake is driven downstream. Once again, the user-specified InflowWind routine evaluates the wind speed acting on the positions required by OpenFAST that, this time, are affected by a certain wake deficit defined according to the model; to perform such a task, the $C_{T,i-1}$ and AIF a_{i-1} averaged on the rotor plane of the first turbine are needed. As before, the obtained values will be passed to the AeroDyn Module of the current OpenFAST instance to obtain the $C_{T,i}$ and $a_{local,i}(r)$. This procedure is carried out for every turbine in the farm at every time step of the simulation. The routine as written gives also the possibility to linearly interpolate the wind input parameters of the simulated wind farm (like the wind speed, direction, surface roughness, etc.).

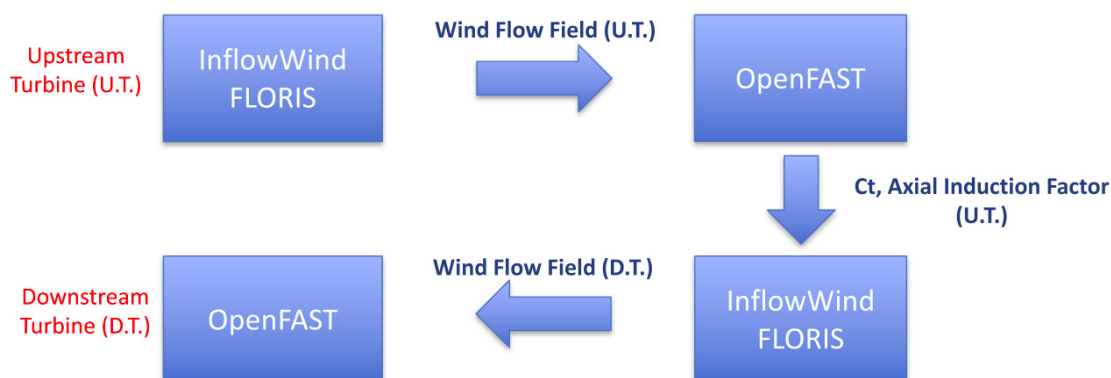


Figure 1. A schematic representation of how the GWM model (the same used by NREL's Floris) is intended to work in the OpenFAST framework.

4. Numerical Setup

In this section, the numerical setup of the OpenFAST simulation, using the implemented model, is proposed; it is extracted from a broader work where the proposed version of the GWM was used to maximize the power extraction of a wind farm by means of a genetic algorithm acting on the yaw angle of the machines (wake-steering optimization method).

4.1. Wind Farm Layout

For the simulation, the reference layout chosen is a 3×1 arrangement of NREL 5MW model turbines, meaning that, having the wind blowing from west, the three turbines are all aligned in that specific direction. These turbines have a diameter $D = 126.0$ m and a hub height $z_h = 90.0$ m. For the coordinates of the hub center point of the turbines, refer to Table 1 or to Figure 2 to graphically visualize the farm and the wake produced by the machines.

Table 1. Reference 3×1 wind farm configuration in absolute and dimensionless values.

| Wind Farm Layout | | | | | | |
|------------------|-------|-------|-------|--------|-----|--------|
| | x (m) | y (m) | z (m) | x/D | y/D | z/D |
| T1 | 0 | 0 | 90 | 0 | 0 | 0.7142 |
| T2 | 700 | 0 | 90 | 5.5555 | 0 | 0.7142 |
| T3 | 1200 | 0 | 90 | 9.5238 | 0 | 0.7142 |

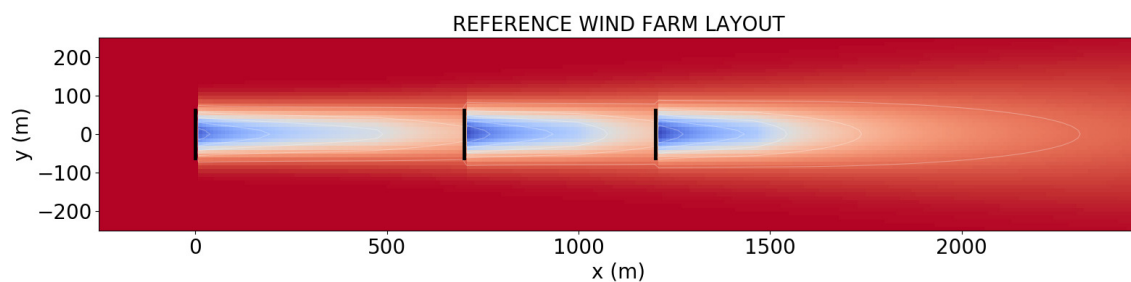


Figure 2. The 3×1 wind farm layout (Image taken from NREL's Floris).

4.2. Wind Model Input

The input parameters of the free-stream logarithmic profile described in the EUROCODE [19], keeping the same notation adopted in Section 2.1, see Table 2:

Table 2. Wind magnitude input for the OpenFAST simulation.

| Wind Magnitude Parameters | | |
|---------------------------|-------|--------|
| V_b | (m/s) | 5.85 |
| z_0 | (m) | 0.005 |
| z_{min} | (m) | 10.0 |
| k_r | - | 0.1617 |
| $u_\infty(z = z_h)$ | (m/s) | 9.2694 |
| $u_\infty(z = z_{min})$ | (m/s) | 7.1900 |
| $I_0(z = z_h)$ | - | 0.1020 |

The wind speed is kept constant during the whole simulation, while the wind direction θ_w changes linearly as described in Table 3:

Table 3. Wind direction input for the OpenFAST simulation.

| Wind Direction Parameters | | | |
|---------------------------|---------------|--------------------------|------------------------|
| t_{start} (s) | t_{end} (s) | $\theta_{w,start}$ (deg) | $\theta_{w,end}$ (deg) |
| 0.0 | 35.0 | 255.0 | 255.0 |
| 35.0 | 70.0 | 255.0 | 270.0 |
| 70.0 | 1300.0 | 270.0 | 270.0 |

For the Bastankhah and Porté-Agel [7,8] GWM, the parameters chosen are reported in Table 4.

Table 4. Bastankhah and Porté-Agel [7,8] GWM parameters for the OpenFAST simulation.

| BPA-GWM Parameters | |
|--------------------|--------|
| k_a | 0.200 |
| k_b | 0.003 |
| α | 0.48 |
| β | 0.077 |
| C_I | 0.8 |
| e_1 | 0.73 |
| e_2 | 0.1 |
| e_3 | −0.275 |

4.3. OpenFAST Settings

For the purposes of the current simulation, the modules of OpenFAST that are used are:

- ElastoDyn, simulating the structural behavior of the machine
- InflowWind, simulating the flow field surrounding the machine
- AeroDyn v15, simulating the aerodynamics of the machine
- ServoDyn, simulating the control part of the machine

On the ElastoDyn input file, the only remark to be made is that all the turbines will start with the low speed rotor spinning at $\dot{\theta}_r = 12.1$ rpm, that is the rated angular speed of the machine; in this case, $u_\infty(z = z_H) < u_{\infty, rated} = 11.4$ m/s, where $u_{\infty, rated}$ is the rated wind speed at which the turbines produce the rated power output, and in the first phase of the simulation, a slow-down of the rotors has to be expected. This means that, even if the wind is taken from a steady-state model, both the structural and aerodynamic part (especially the C_T and the AIF that are the most important part of the current model) of the turbine will need a certain amount of time to reach the equilibrium condition; so, to obtain accurate results for a given simulation, about 50.0 s are required when the wind conditions are changed.

Moreover, in the ElastoDyn Module, the yaw DOF was activated; it was used to apply wake steering in order to increase the power extraction of the wind farm. The change of the yaw angle of turbines T1 and T2 is the reason why, in the wind speed graphs shown in Figures 3–5, the wind magnitude in the x -axis direction of the reference wind farm layout is continuously changing. However, this particular optimization method shows in a complete way how the proposed wake model works in the wind farm.

In the InflowWind Module, the model of wind used was the user-specified routine presented in Section 2.4, the one given by the Bastankhah and Porté-Agel [7,8] GWM model.

In the AeroDyn v15 Input File, it was necessary to set some parameters for the aerodynamic simulation of the machine (BEMT model). Since the wind model employed is stationary, a classic BEMT model was used. The maximum number of iterations allowed for the convergence of the AIF and TIF (activated in the simulation) was fixed to $n_{it} = 10,000$, in order to be sure to reach convergence with the tolerance set to the OpenFAST “default” value. Glauert’s tip loss and hub loss corrections were used in the simulation. For the blades airfoil aerodynamics, the unsteady Beddoes–Leishman

model with the Minnema–Pierce variant was used. Finally, to take into account the effect that the skewed wake has on the yawed turbine, the Pitt and Peters BEMT correction was activated.

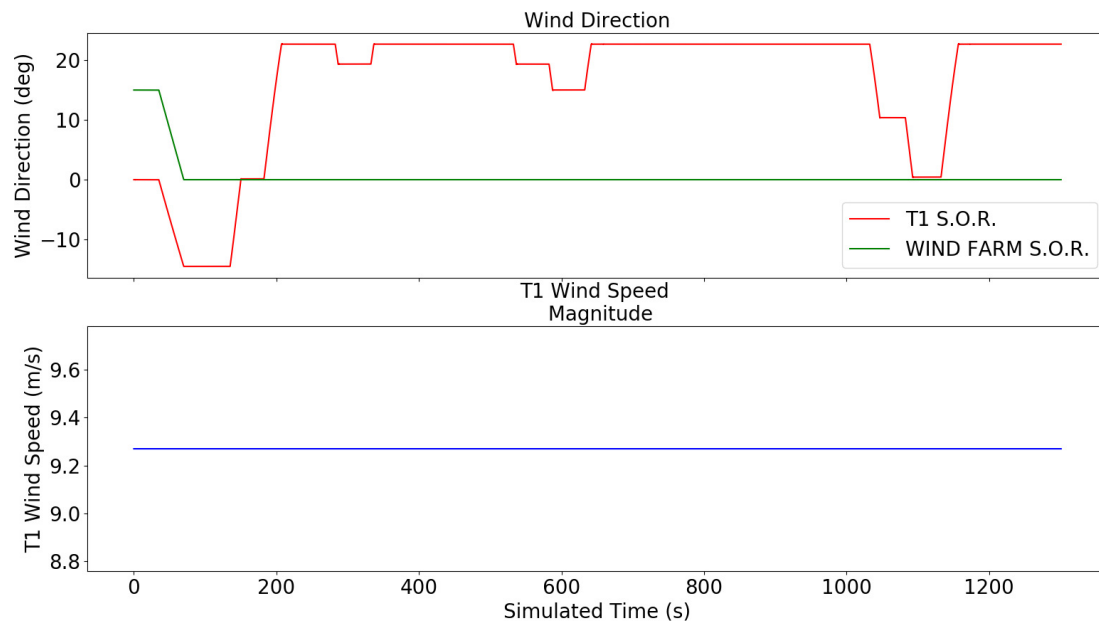


Figure 3. Wind speed and direction acting on the hub of turbine T1 during the simulation.

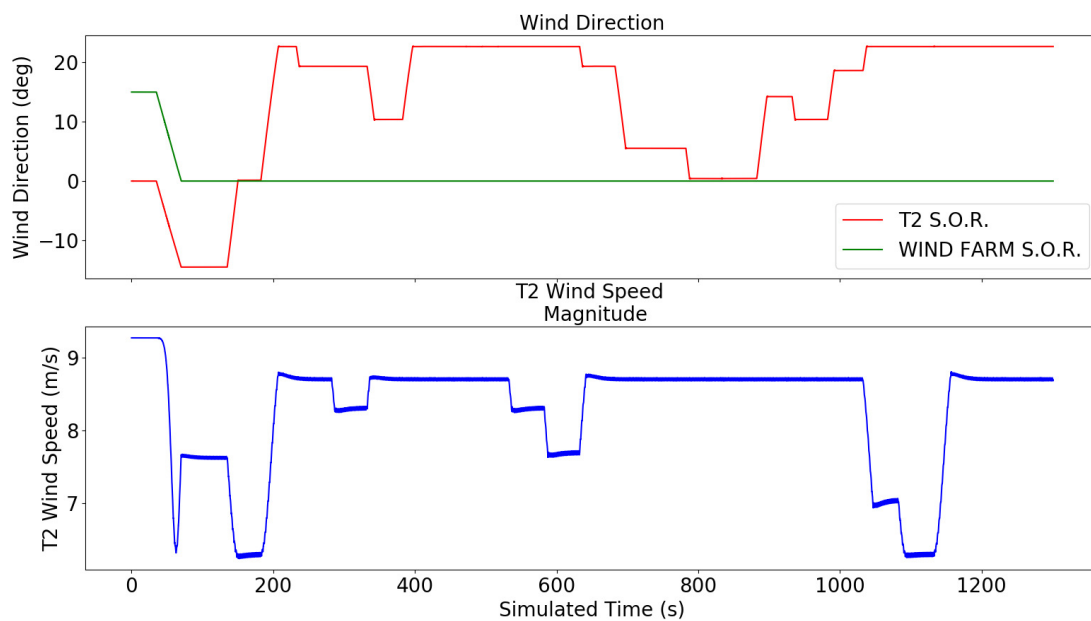


Figure 4. Wind speed and direction acting on the hub of turbine T2 during the simulation.

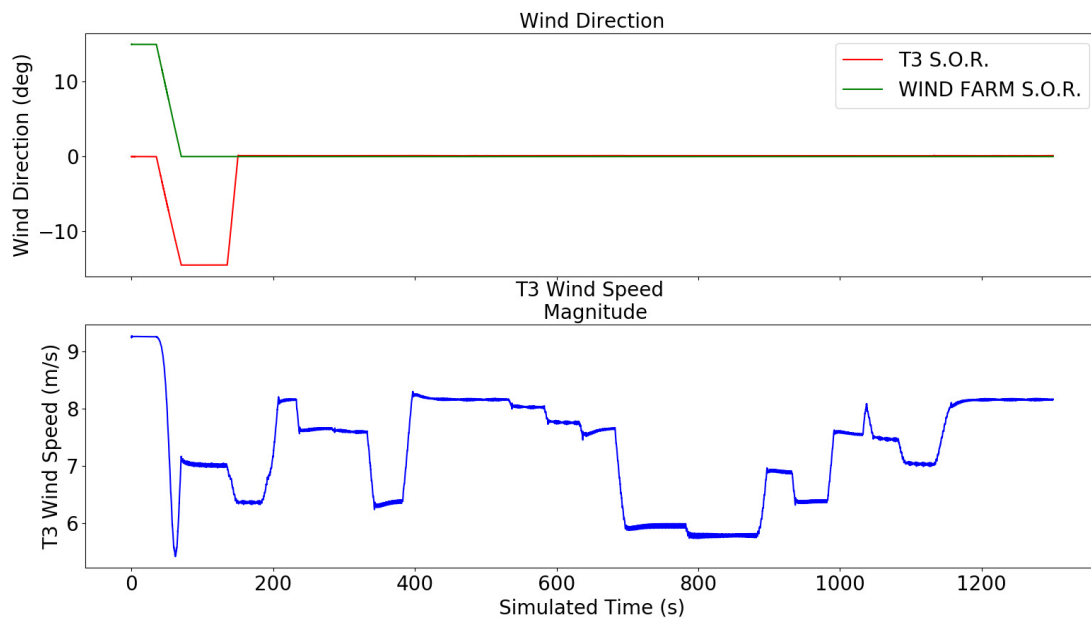


Figure 5. Wind speed and direction acting on the hub of turbine T3 during the simulation.

5. Results and Discussion

In this section, the results obtained from the simulation will be presented, focusing mainly on the wind model, together with a validation of the OpenFAST/Floris code against NREL's Floris.

5.1. OpenFAST/Floris Validation Results

The modified model was validated against the Floris software to be sure of a correct implementation. The standard Floris version was slightly changed in order to allow for the evaluation of the added turbulence I_+ using an AIF specified as an added input from the user. This was possible since the AIF value is never explicitly used in the calculations of the wake deficit and deflection except when it comes to the added turbulence intensity (the AIF value is obtained as a function of the C_T as in Equation (8)). This change allows for inserting the AIF obtained from the OpenFAST simulation to perform a correct validation.

The validation was performed using the same numerical setup of the presented simulation, but converting the parameters of the EUROCODE [19] logarithmic profile into an equivalent power law (since Floris uses this model of free wind). The wind values presented in Table 5 make reference to the hub position and height of the three turbines, while the C_T and C_P are synthetic parameters referring to the whole rotor plane.

The difference between the codes is:

$$e = \frac{v_{act} - v_{fl}}{v_{fl}} * 100 \quad (33)$$

The slight deviation from the Floris software results is limited and is expected due to the proposed changes. A proper consideration of rotor aerodynamics increases the wind speed at the hub height for the turbines in the wake for the considered case, increasing the performance of the wind farm.

5.2. Simulation Results

Making reference to Figures 3–5, it is possible to understand how the GWM proposed by Bastankhah and Porté-Agel [7,8], together with the additional changes described in Section 2.3,

affects the inflow wind felt by each downstream turbine (T2 and T3, named as in Figure 2). The various fluctuations in the graphs of the wind speed acting on the hub of the various turbines are justified by the fact that, during the simulation, the machines are continuously yawing (in particular, T1 and T2), changing in this way the wake propagated downstream.

To better understand this effect, the yaw angles of T1 and T2 are reported in Figure 6. For the upstream turbine T1, the wind felt is always constant, while the downstream ones feel a different wind when the yaw angles of T1 and T2 are changed according to the wind farm control logic used in the simulation. The yaw angle of turbine T3 is always aligned in the wind direction, since this is the only position that maximizes the total power extraction. For example, referring to Figures 4 and 6, respectively the wind speed felt at turbine T2's hub and the yaw angle of T1 during the simulation, it can be noted that for $t_{sim} \in [1000 - 1150]$ s, the yaw angle of T1 is passing from -25 degrees to -10 degrees and finally to 0 degrees. At the same time, the wind speed of T2, due to the steered wake of T1, is passing from $\simeq 9$ m/s ($\gamma_{T1} = -25$ deg) to $\simeq 7$ m/s ($\gamma_{T1} = -10$ deg) to $\simeq 6$ m/s ($\gamma_{T1} = 0$ deg), because the wake produced by T1 is gradually overlapping T2's rotor plane.

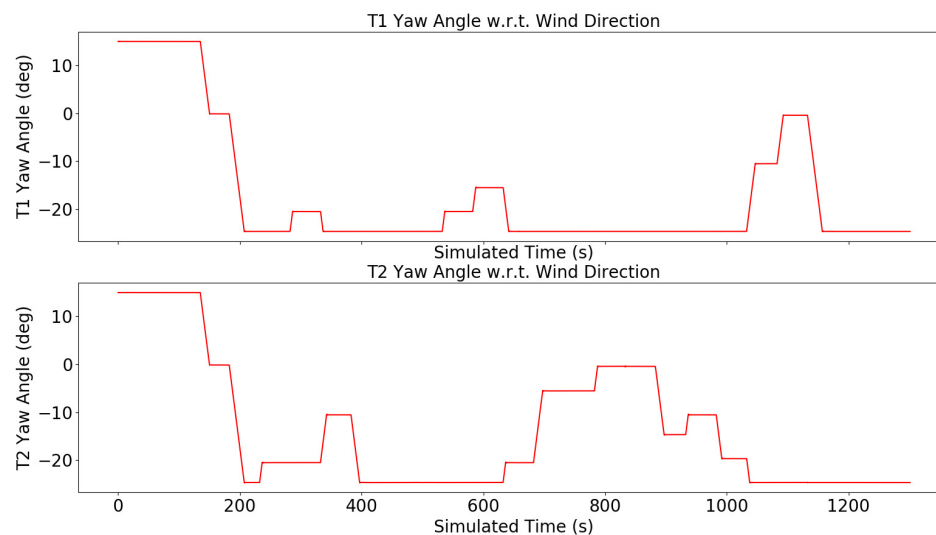


Figure 6. Yaw angle of turbines T1 and T2 during the simulation.

Table 5. OpenFAST/Floris validation.

| OpenFAST/Floris Validation | | | |
|----------------------------|--------------|--------------|-----------|
| Wind Speed | | | |
| | ACTUAL (m/s) | FLORIS (m/s) | ERROR (%) |
| T1 | 9.2694 | 9.2694 | 0.0 |
| T2 | 6.3 | 5.8318 | 8.028 |
| T3 | 6.36 | 5.8579 | 8.571 |
| C_p | | | |
| | ACTUAL (-) | FLORIS (-) | ERROR (%) |
| T1 | 0.4655 | 0.4684 | -0.6191 |
| T2 | 0.5225 | 0.4611 | 13.315 |
| T3 | 0.5050 | 0.4583 | 10.189 |
| C_T | | | |
| | ACTUAL (-) | FLORIS (-) | ERROR (%) |
| T1 | 0.7525 | 0.7448 | 1.0338 |
| T2 | 0.8140 | 0.8004 | 1.6991 |
| T3 | 0.8100 | 0.8090 | 0.1236 |

The same analysis can be done for other time periods of the simulation and considering different combinations of turbines, like T2 and T3, but paying attention to the fact that in this case, also T1's wake is contributing to the definition of the wind speed felt at T3's hub. As reported in Sections 2.1 and 2.4, the model is strongly dependent on the thrust coefficient C_T and on the rotor averaged AIF a . From the graphs in Figures 7 and 8, it is possible to see the trend of the C_T and a of the various turbines and understand how the wind velocities reflect them.

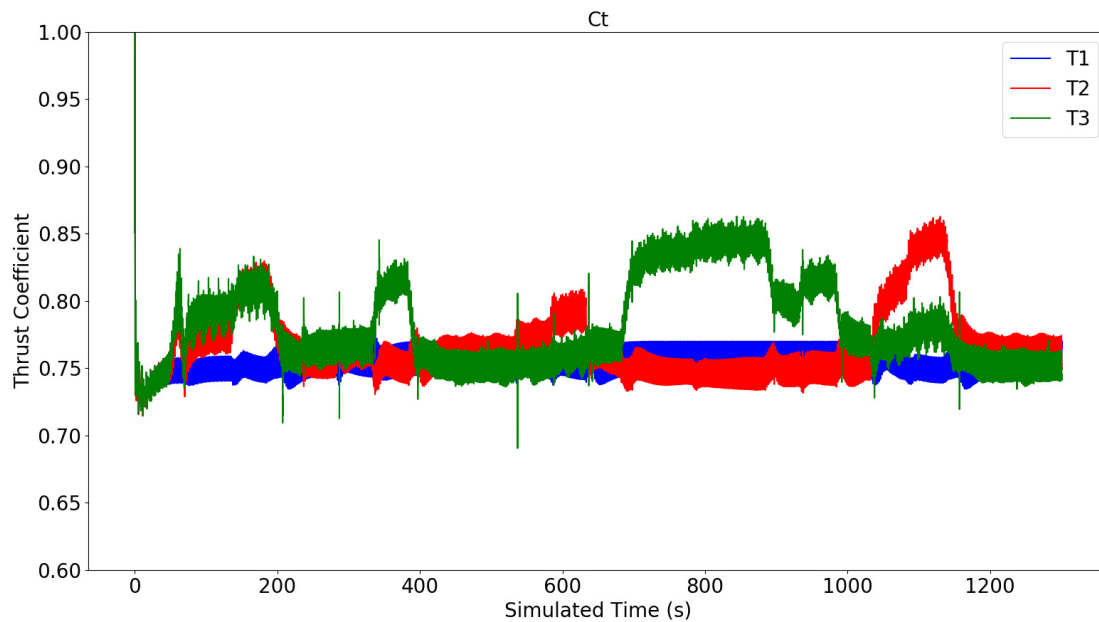


Figure 7. The C_T of the three NREL 5MW turbines during the simulation.

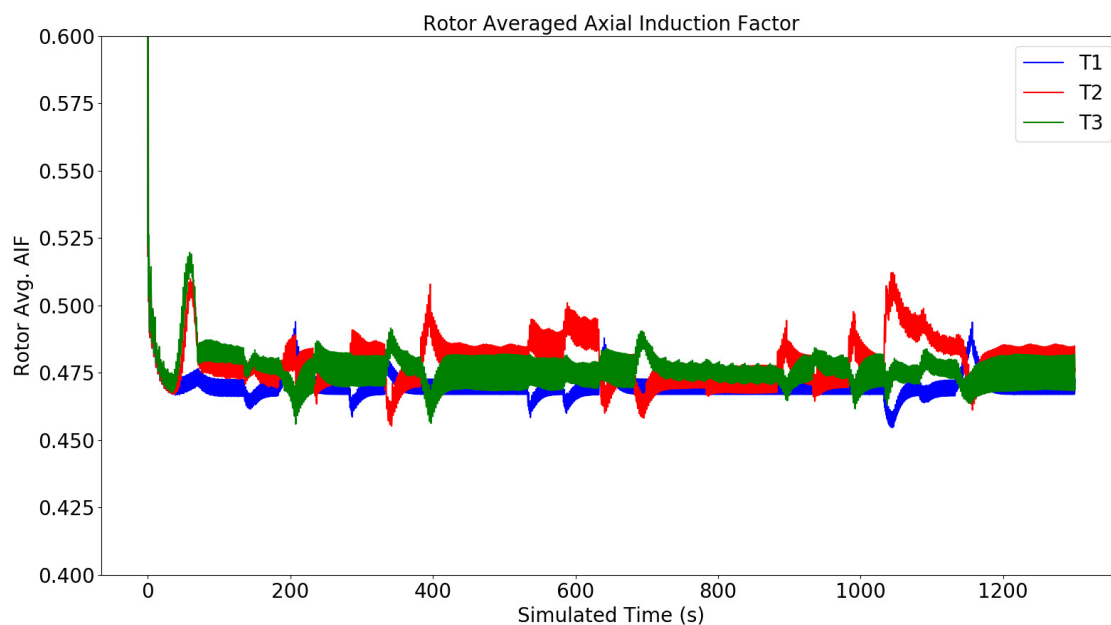


Figure 8. The rotor averaged AIF a of the three NREL 5MW turbines during the simulation.

5.3. Discussion

The implementation in the OpenFAST framework of the Bastankhah and Porté-Agel [7,8] GWM, together with the additional models reported in Section 2.3 allow using the OpenFAST software not only as a single turbine aero-elastic simulator, but also as a complete, standalone simulator for an entire wind farm with the hypothesis of steady-state environmental conditions. The simulation speed of this new tool could further be increased if all the OpenFAST instances (each one representing a single turbine) were run in parallel fashion; however, this would mean rethinking and rewriting the whole InflowWind user-specified routine.

Since the implemented model is a steady-state model, this new tool could not be used in order to simulate dynamic or transient conditions, and thus to study wind farm control techniques relying on wake dynamics. This is the major drawback of the model, together with the impossibility of reproducing adequately the wake meandering phenomenon that has quite a big effect on the flow field developed inside a wind farm. In the future, it could be possible to add wake meandering to the model by using the superposition principle. Another limitation of the present model is the need to set up the four parameters k_a , k_b , α , and β according to how the wakes develop in the wind farm under consideration. Estimating these parameters requires an ad hoc study campaign that could be useful also to validate the presented model with field data.

6. Conclusions and Future Work

In this paper, some changes made to the Bastankhah and Porté-Agel [7,8] GWM, in order to implement it in the OpenFAST framework, are presented. Moreover, the organization and the use of the InflowWind routine is explained. After this, in order to verify how the model behaves, a simulation is carried out using a reference 3×1 wind farm layout; the results show that the wake model works as intended (since the results obtained are validated also using the same wind inputs and wind farm layout on the Floris software). Finally, Section 5.3 shows the purposes for which the model could be used and the pros and cons that this tool carries with it when compared to other simulation methods.

In the future, to enhance the capabilities of the OpenFAST/GWM coupling, a wake transport model has to be added in order to correctly reproduce the downstream propagation of the wave. This is the last fundamental step for the creation of an advanced tool able to analyze the behavior of a wind farm (both in terms of loads and power extraction) under a mean flux condition. With a further refinement of the code, the developed software could be used as an advanced Floris (control-oriented model) able to perform calculations in real time. Introducing these two changes, it is possible to obtain the accuracy of a medium fidelity model with the computational speed of a low fidelity control-oriented model. From an industrial standpoint, refining the code introducing newer features like a wake transport and a turbulence model could lead to the creation of a tool that wind farm managers can use to perform real-time multi-objective optimization tasks (i.e., maximize the power extraction of the farm while keeping the loads acting on the blade as low as possible) or as an observer able to estimate what is happening in real time in the farm using only a few measurements from the turbines.

In the next months, a custom and open-source version of OpenFAST containing the presented model will be released.

Author Contributions: Conceptualization, P.S.; formal analysis, A.C.; investigation, A.C.; resources, A.Z.; supervision, C.M. and P.S.; writing, review and editing, C.M. and P.S. All authors read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Acknowledgments: Thanks to the National Renewable Energy Laboratory (NREL) for the open-source software that was used in this work.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

| | |
|------------|---------------------------------------|
| θ_r | Turbine's rotor angular speed |
| GWM | Gaussian Wake Model |
| C_T | Thrust coefficient |
| C_P | Power coefficient |
| AIF | Axial Induction Factor |
| a | Axial induction factor |
| ABL | Atmospheric Boundary Layer |
| ALM | Actuator Line Model |
| API | Application Programming Interface |
| BEMT | Blade Element Momentum Theory |
| DBEMT | Dynamic Blade Element Momentum Theory |
| DOF | Degree Of Freedom |
| DWM | Dynamic Wake Meandering |
| RANS | Reynolds Averaged Navier–Stokes |
| S.o.R. | System of Reference |
| TIF | Tangential Induction Factor |
| TSR | Tip Speed Ratio |

Appendix A. InflowWind Routine Description

The main routine called **IfW_UserWind_CalcOutput** is divided into sub-functions, each performing a smaller task:

- **WindCalc**,
responsible for the evaluation of the wind speed $\bar{u}(x, y, z)$ and the wake deflection δ at a given point (x, y, z) asked from OpenFAST;
- **IntersectArea**,
responsible for the evaluation of the intersection area between two given turbines (it determines the $A_{ol,\%}$ term appearing in Equation (21));
- **RotoTransPoint**,
responsible for performing rotations and translations of a point from one S.o.R. to another (i.e., from a non-yawed rotor plane to a yawed rotor plane);
- **AddedTurb**,
responsible for the evaluation of the added turbulence intensity generated in a wake (the I_+ term appearing in Equation (21));
- **RotLayoutAndSort**,
responsible for performing rotations and translations of the layout of the wind farm to align the turbines in wind direction (in order to perform the calculations needed to obtain the wind speed and deflection at a given point) and to sort the turbines from the most upstream one to the last downstream one along the stream-wise direction;
- **RotAvgAxInd**,
responsible for the evaluation of the rotor averaged AIF (from the local values $a_{local}(r)$ on the blades, obtaining only one averaged value of a).

To pass from $a_{local}(r)$ to a , a weighted averaged based on the distance from the rotor's hub is performed, like in the following equation:

$$a = \frac{1}{N_{blades} \sum_{i=1}^{N_p} r_i} \sum_{j=1}^{N_{blades}} \sum_{i=1}^{N_p} a_{local,j,i} r_{j,i} \quad (A1)$$

where the j index represents the number of blades for the considered turbine, the i index represents the number of points considered for the blade, and $r_{j,i}$ represents the distance from the hub for the section to which $a_{local,j,i}$ makes reference.

All these subroutines are used in **IfW_UserWind_CalcOutput** to fully characterize the wind speed acting on a given point of the rotor of each turbine simulated in the wind farm at every time step.

References

1. Van Kuik, G.A.M.; Peinke, J.; Nijssen, R.; Lekou, D.; Mann, J.; Sorensen, J.N.; Ferreira, C.; van Wingerden, J.W.; Schlipf, D.; Gebraad, P.; et al. Long-term research challenges in wind energy—A research agenda by the European Academy of Wind Energy. *Wind Energy Sci.* **2016**, *1*, 1–39. [CrossRef]
2. NREL. *OpenFAST*; NREL: Golden, CO, USA, 2020. Available online: <https://openfast.readthedocs.io/en/master/> (accessed on 20 October 2020).
3. NREL. *SOWFA*; NREL: Golden, CO, USA, 2020. Available online: <https://www.nrel.gov/wind/nwtc/sowfa.html> (accessed on 20 October 2020).
4. OpenFOAM Foundation. *OpenFOAM*. 2020. Available online: <https://openfoam.org/> (accessed on 20 October 2020).
5. Schito, P.; Zasso, A. Actuator forces in CFD: RANS and LES modeling in OpenFOAM. *J. Phys. Conf. Ser.* **2014**, *524*, 012160. [CrossRef]
6. NREL. *FLORIS Version 2.2.0*; NREL: Golden, CO, USA, 2020. Available online: <https://www.nrel.gov/wind/assets/pdfs/systems-engineering-workshop-2019-floris.pdf> (accessed on 20 October 2020).
7. Bastankhah, M.; Porté-Agel, F. Experimental and theoretical study of wind turbine wakes in yawed conditions. *J. Fluid Mech.* **2016**, *806*, 506–541. [CrossRef]
8. Bastankhah, M.; Porté-Agel, F. A new analytical model for wind-turbine wakes. *Renew. Energy* **2014**, *70*, 116–123. [CrossRef]
9. Larsen, G.C.; Aagaard, H.M.; Bingöl, F.; Mann, J.; Ott, S.; Sørensen, J.N.; Okulov, V.; Troldborg, N.; Nielsen, N.M.; Thomsen, K.; et al. *Dynamic Wake Meandering Modeling*; Risø National Laboratory: Roskilde, Denmark, 2007.
10. van Dijk, M.T.; van Wingerden, J.W.; Ashuri, T.; Li, Y. Wind farm multi-objective wake redirection for optimizing power production and loads. *Energy* **2017**, *121*, 561–569. [CrossRef]
11. Bay, C.; King, J.; Fleming, P.; Mudafort, R.; Martínez Tossas, L. Unlocking the Full Potential of Wake Steering: Implementation and Assessment of a Controls-Oriented Model. *Wind Energy Sci. Discuss.* **2019**, 1–20. [CrossRef]
12. Gebraad, P.; Fleming, P.; Wright, A.; Dykes, K.; van Wingerden, J.W. *Multidisciplinary Research on Wake Control in Wind Power Plants at NREL*; NREL: Golden, CO, USA, 2015.
13. Bossanyi, E. Combining induction control and wake steering for wind farm energy and fatigue loads optimisation. *J. Phys. Conf. Ser.* **2018**, *1037*, 032011. [CrossRef]
14. Jenkins, N.; Burton, A.; Sharpe, D.; Bossanyi, E. *Wind Energy Handbook*; John Wiley & Sons Ltd.: London, UK, 2001.
15. Lee, J.H.; Chu, V. *Turbulent Jets and Plumes*; Springer: Berlin/Heidelberg, Germany, 2012.
16. Katic, I.; Højstrup, J.; Jensen, N. A Simple Model for Cluster Efficiency. In Proceedings of the European Wind Energy Association Conference and Exhibition, Rome, Italy, 6–8 October 1986; Volume 1, pp. 407–410.
17. Crespo, A.; Hernandez, J. Turbulence characteristics in wind-turbine wakes. *J. Wind Eng. Ind. Aerodyn.* **1996**, *61*, 71–85. [CrossRef]
18. Niayifar, A.; Porté-Agel, F. A new analytical model for wind farm power prediction. *J. Phys. Conf. Ser.* **2015**, *625*, 012039. [CrossRef]
19. Eurocode 1: *Actions on Structures—Part 1–4: General Actions—Wind Actions*; European Committee for Standardization: Bruxelles, Belgium, 2005.

Publisher’s Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).